THE UNIVERSITY
of EDINBURGH

This thesis has been submitted in fulfilment of the requirements for a postgraduate degree (e.g. PhD, MPhil, DClinPsychol) at the University of Edinburgh. Please note the following terms and conditions of use:

This work is protected by copyright and other intellectual property rights, which are retained by the thesis author, unless otherwise stated.
A copy can be downloaded for personal non-commercial research or study, without prior permission or charge.
This thesis cannot be reproduced or quoted extensively from without first obtaining permission in writing from the author.
The content must not be changed in any way or sold commercially in any format or medium without the formal permission of the author.
When referring to this work, full bibliographic details including the author, title, awarding institution and date of the thesis must be given.

Overcoming the limitations of statistical parametric speech synthesis

Thomas Merritt

Doctor of Philosophy
Institute for Language, Cognition and Computation
School of Informatics
University of Edinburgh
2016

Lay Summary

Speech synthesis is the process of generating speech for given text, hence is commonly referred to as text-to-speech (TTS). Speech synthesis has many useful applications such as; providing a means of interaction with dialog systems, automatic production of audio books or for use as a voice for those who are unable to speak due to surgery. There are two common approaches to TTS; using real examples of speech cut up into 'sections' of speech, which are then stuck back together to generate new speech (referred to as unit selection), or building models of speech using machine learning methods to generate speech. Each of these methods has their respective advantages. Unit selection produces extremely natural speech provided the required 'sections' of speech are available (as these examples are natural recordings). Where this isn't the case synthesis quality drops dramatically due to large artefacts between these examples. Modelled speech, however, has lower overall quality yet is able to generate for unseen speech much better, making it more flexible.

At the time of beginning this thesis, statistical parametric speech synthesis (SPSS) using hidden Markov models (HMMs) was the dominant synthesis paradigm within the research community. SPSS systems are effective at generalising across the linguistic contexts present in training data to account for the inevitable unseen linguistic contexts at synthesis-time, making these systems flexible and their performance stable. However HMM synthesis suffers from a 'ceiling effect' in the naturalness achieved meaning that, despite great progress, the speech output is rarely confused for natural speech. There are many hypotheses for the causes of reduced synthesis quality, and subsequent required improvements, for HMM speech synthesis in literature. However, until this thesis, these hypothesised causes were rarely tested.

This thesis makes two types of contributions to the field of speech synthesis; each of these appears in a separate part of the thesis. Part I introduces a methodology for testing hypothesised causes of limited quality within HMM speech synthesis systems. This investigation aims to identify what causes these systems to fall short of natural speech. Part II uses the findings from Part I of the thesis to make informed improvements to speech synthesis.

The usual approach taken to improve synthesis systems is to attribute reduced synthesis quality to a hypothesised cause. A new system is then constructed with the aim of removing that hypothesised cause. However this is typically done without prior testing to verify the hypothesised cause of reduced quality. As such, even if improvements

in synthesis quality are observed, there is no knowledge of whether a real underlying issue has been fixed or if a more minor issue has been fixed. In contrast, I perform a wide range of perceptual tests in Part I of the thesis to discover what the real underlying causes of reduced quality in HMM synthesis are and the level to which they contribute.

Using the knowledge gained in Part I of the thesis, Part II then looks to make improvements to synthesis quality. Two well-motivated improvements to standard HMM synthesis are investigated. The first of these improvements follows on from averaging across differing linguistic contexts being identified as a major contributing factor to reduced synthesis quality. This is a practice typically performed during decision tree regression in HMM synthesis. Therefore a system which removes averaging across differing linguistic contexts and instead performs averaging only across matching linguistic contexts (called rich-context synthesis) is investigated. The second of the motivated improvements follows the finding that the parametrisation (i.e., vocoding) of speech, standard practice in SPSS, introduces a noticeable drop in quality before any modelling is even performed. Therefore the hybrid synthesis paradigm is investigated. These systems aim to remove the effect of vocoding by using SPSS to inform the selection of units in a unit selection system.

Abstract

At the time of beginning this thesis, statistical parametric speech synthesis (SPSS) using hidden Markov models (HMMs) was the dominant synthesis paradigm within the research community. SPSS systems are effective at generalising across the linguistic contexts present in training data to account for inevitable unseen linguistic contexts at synthesis-time, making these systems flexible and their performance stable. However HMM synthesis suffers from a 'ceiling effect' in the naturalness achieved, meaning that, despite great progress, the speech output is rarely confused for natural speech. There are many hypotheses for the causes of reduced synthesis quality, and subsequent required improvements, for HMM speech synthesis in literature. However, until this thesis, these hypothesised causes were rarely tested.

This thesis makes two types of contributions to the field of speech synthesis; each of these appears in a separate part of the thesis. Part I introduces a methodology for testing hypothesised causes of limited quality within HMM speech synthesis systems. This investigation aims to identify what causes these systems to fall short of natural speech. Part II uses the findings from Part I of the thesis to make informed improvements to speech synthesis.

The usual approach taken to improve synthesis systems is to attribute reduced synthesis quality to a hypothesised cause. A new system is then constructed with the aim of removing that hypothesised cause. However this is typically done without prior testing to verify the hypothesised cause of reduced quality. As such, even if improvements in synthesis quality are observed, there is no knowledge of whether a real underlying issue has been fixed or if a more minor issue has been fixed. In contrast, I perform a wide range of perceptual tests in Part I of the thesis to discover what the real underlying causes of reduced quality in HMM synthesis are and the level to which they contribute.

Using the knowledge gained in Part I of the thesis, Part II then looks to make improvements to synthesis quality. Two well-motivated improvements to standard HMM synthesis are investigated. The first of these improvements follows on from averaging across differing linguistic contexts being identified as a major contributing factor to reduced synthesis quality. This is a practice typically performed during decision tree regression in HMM synthesis. Therefore a system which removes averaging across differing linguistic contexts and instead performs averaging only across matching linguistic contexts (called rich-context synthesis) is investigated. The second of the motivated improvements follows the finding that the parametrisation (i.e., vocoding) of speech, standard practice in SPSS, introduces a noticeable drop in quality before any

modelling is even performed. Therefore the hybrid synthesis paradigm is investigated. These systems aim to remove the effect of vocoding by using SPSS to inform the selection of units in a unit selection system. Both of the motivated improvements applied in Part II are found to make significant gains in synthesis quality, demonstrating the benefit of performing the style of perceptual testing conducted in the thesis.

Acknowledgements

I would like to thank the following people:

- Simon King for all his supervision, help and guidance.
- Rob Clark and Junichi Yamagishi for all of their feedback, help and discussion of new ideas. In addition, thanks to Rob Clark for implementing hybrid synthesis into the Festival Multisyn code.
- Gustav Eje Henter for providing the code for the MUSHRA testing and analysis used in this thesis.
- Zhizheng Wu for providing me with the code to implement the DNN synthesis systems used in this thesis.
- All of my collaborators; Gustav Eje Henter, Javier Latorre, Cassie Mayo, Tuomo Raitio, Srikanth Ronanki, Matt Shannon, Oliver Watts and Zhizheng Wu. Our discussion of ideas and your helpful feedback contributed to increasing my understanding of the field of research.
- Rosie Kay, Alexandra Delipalta and Michael Hobart for their great help recruiting listeners and overseeing the running of perceptual tests.
- Cássia Valentini Botinhão for her help proof reading my thesis.
- Avril Heron for her help binding my thesis.
- The researchers on the NST project for our discussion of ideas.
- Steve Renals for his feedback during my annual reviews.
- Everyone in CSTR for helping with feedback on my work and for making the last few years so enjoyable.
- Harriet O'Rourke for her patience and support throughout my PhD.
- My family for all their support.

This research was made possible by EPSRC Programme Grant EP/I031022/1, Natural Speech Technology (NST).

Declaration

I declare that this thesis was composed by myself, that the work contained herein is my own except where explicitly stated otherwise in the text, and that this work has not been submitted for any other degree or professional qualification except as specified.

(*Thomas Merritt*)

Table of Contents

List of Figures

List of Tables

Notation

The following notation has been used throughout this thesis.

ASF	acoustic space formulation
BAP	band-aperiodicities
DCT	discrete cosine transform
DFT	discrete Fourier transform
DNN	deep neural network
DTW	dynamic time warping
EM	expectation maximisation
f_0	fundamental frequency
GMM	Gaussian mixture model
GV	global variance
HMM	hidden Markov model
HSMM	hidden semi-Markov model
HTK	hidden Markov model toolkit
HTS	HMM-based speech synthesis system
KLD	Kullback-Leibler divergence
LPC	linear predictive coding
LSF	line spectral frequency
LTS	letter-to-sound
MCD	Mel cepstral distortion
MDL	minimum description length
MDS	multidimensional scaling
MFCC	Mel-frequency cepstral coefficients
MLPG	maximum likelihood parameter generation
MOS	mean opinion score
MUSHRA	MUltiple Stimuli with Hidden Reference and Anchor
MSD	multi-space probability distribution

POS	part of speech
RNN	recurrent neural network
SPSS	statistical parametric speech synthesis
SPTK	speech signal processing toolkit
TTS	text-to-speech

Chapter 1

Background

1.1 Introduction

Speech synthesis, usually referred to as text-to-speech (TTS), is the process of generating speech from a given text. This is an important technology which has many useful applications, such as providing a tool for interaction with dialogue systems, or for use as a voice for those who are unable to speak (Veaux et al., 2012).

Statistical parametric speech synthesis (SPSS) is the dominant synthesis paradigm within the TTS research community. In SPSS speech sounds from training data are modelled. At synthesis-time, the models are used to predict required speech sounds. The main SPSS approach at the time of beginning work on this thesis was to use hidden Markov models (HMMs). The HMM paradigm offers flexible synthesis, by generalising well across the speech present in the training data in order to account for the inevitable speech sounds which were unseen in the training data. HMM synthesis systems are also able to produce very intelligible speech. However for maximum effectiveness in engaging with the listener, the speech produced from a TTS system should also sound natural, i.e., believable as a human speaker. This needs to be the case across the vast number of possible different permutations of speech sounds which could be required. Thus far, this is a criterion which current SPSS systems fall short on, as consistently highlighted in successive results from the annual Blizzard Challenge (King and Karaiskos, 2009, 2010, 2011, 2012, 2013). Causes of the loss in quality, relative to synthesis using waveform concatenation, experienced in SPSS have been hypothesised in the literature, however until work began on this thesis these hypothesised causes were rarely formally tested. Therefore this thesis will focus on formally testing hypothesised causes of reduced quality within SPSS. These findings will then

provide insight into which areas of SPSS require improvements, allowing for informed improvements to be made.

As the research approach taken in this thesis comprises of two different (but complimentary) aims, these are reflected in the two different parts of the thesis. Part I focuses on formally testing hypothesised causes of reduced quality in SPSS, investigating not only which elements of current SPSS systems have a detrimental impact on synthesis but also quantifying the effect of each of these. Part II of the thesis then uses the findings from the investigations in Part I to make informed improvements to TTS synthesis.

1.2 Chapter overview

This chapter provides an overview of the field of speech synthesis to a sufficient level for understanding the work in Part I of the thesis. Part I of the thesis acts on the state of the art speech synthesis research at the time of beginning this thesis, i.e., decision tree-based HMM speech synthesis. Following the start of work on this thesis there was a resurgence of research into using deep neural networks (DNNs) for speech synthesis. Such updates in speech synthesis research, between beginning work on Part I of the thesis and beginning work on Part II of the thesis, will be discussed in Chapter 8.

1.3 Text-To-Speech

The process of converting text to synthesised speech (TTS) is composed of two major components, often referred to as the front-end and the back-end. The front-end of a TTS system takes the text to be synthesised and converts this into its linguistic components. The back-end of the TTS system then uses these linguistic components to generate a speech waveform.

1.3.1 Front-End

The front-end of a TTS system takes the raw text as characters and converts these to a linguistically-meaningful representation of the text. From such a linguistic representation it is then possible to classify elements of speech. Due to the nature of this element of the synthesis pipeline, the front-end is heavily language-specific.

First text normalisation is performed to convert non-standard words, such as numerical items, into text. Part of speech (POS) tagging of the word sequence is performed. The lexicon (also known as the dictionary) is used to convert the words into sequences of phonemes. If a word is not present in the lexicon, letter-to-sound (LTS) rules are used to predict the phoneme sequence. Phrase breaks in the utterance are predicted. Finally post-lexical rules are applied. Linguistic context features from the front-end processing are passed to the synthesis back-end to produce speech. The front-end used in this thesis is Festival (Black et al., 2001) (the standard front-end for HTS). A combination of the features extracted by the front-end are stored for use in the synthesis back-end. The combination of these features are referred to as a linguistic context. The features from the front-end, which are used in the synthesis back-end in this thesis are listed on page 1043 of Zen et al. (2009).

1.3.2 Back-End

The back-end of the TTS system takes the language-specific output from the front-end and uses this to produce a speech waveform. The back-end of the synthesis system is much less language-dependant, because the language processing within the system has already been performed in the front-end. The fundamental approaches taken by the back-end of TTS systems can be split into two different methods: exemplar-based synthesis and model-based synthesis (Taylor describes how these two approaches can in fact be explained in one unified framework in Taylor (2006b)). These methods will be explained in Sections 1.4 & 1.5.

The work in this thesis focuses on the back-end of the TTS pipeline, however it is perfectly reasonable to assume that a similar style of investigative research is possible focused instead on the front-end. Given that the front-end and back-end components are independent but complimentary in the synthesis pipeline it is reasonable to predict that improvements made to the front-end should also provide additional gains to the naturalness of the speech output from the TTS synthesis system (Kay et al., 2015).

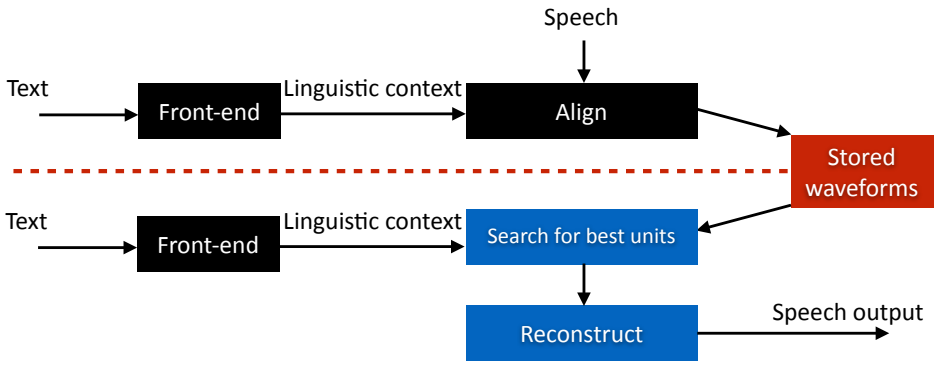


Figure 1.1: *Illustration of the unit selection speech synthesis pipeline. Above the red dotted line refers to the training phase and below the line refers to the synthesis phase.*

1.4 Exemplar-based synthesis

Exemplar-based synthesis systems use real examples of speech to synthesise unseen sentences. At training-time, exemplar-based synthesis systems store natural examples of speech from the training data. These are stored alongside the linguistic contexts, as output from the front-end of the synthesis system, which describe the content of each speech example. These pairings of linguistic contexts and speech samples are stored in the unit database. In many cases the waveform signal itself is stored, otherwise a representation such as linear predictive coding (LPC) may be used. The unit selection pipeline is shown in Figure 1.1.

At synthesis-time, the text to be synthesised is passed through the front-end to provide the linguistic contexts which the utterance comprises. These linguistic contexts then provide the target from which to search for speech examples from the unit database. There are two main exemplar-based synthesis systems: diphone synthesis and unit selection synthesis. Diphone synthesis evolved into unit selection synthesis as computational capacity increased (Taylor, 2009, pages 474–477). These two types of systems handle synthesis in slightly different ways, reflecting the computational power and storage available at the different times.

The domain of exemplar-based speech synthesis began with diphone synthesis (Moulines and Charpentier, 1990). Diphone synthesis gets its name from the unit size used for synthesis. The diphone unit size is a popular choice as the middle of a phone is relatively stable making it suitable for smooth joining of units (Taylor, 2009, page 401). The middle of a phone is more stable than a phone boundary due to co-

articulation. Also, if there is an error in the annotation of a phone boundary this is more likely to result in artefacts being present in the synthesised speech for phone-sized units than for diphone-sized units. In diphone synthesis there is generally one example of each diphone in the unit database. The chances of a smooth transition between speech units without performing modifications are therefore very slim. As a result, when the candidate matching the diphone identity of the target diphone is selected, modifications are made to the candidate in order to better suit the desired target f_0 and duration. Due to the modifications performed, the resulting synthesised speech is of low quality. Diphone synthesis was in mainstream use when far less computational power and storage was available. As such this method of synthesis was at the cutting edge of what was possible at the time. Diphone synthesis is still used in embedded systems where larger amounts of storage are not feasible (Sündermann et al., 2005).

Exemplar-based speech synthesis then developed from diphone synthesis to unit selection synthesis as larger amounts of storage space and quicker search times became possible. Unit selection is an extension of diphone synthesis in which multiple examples of each unit type are used. The increased examples of linguistic contexts allows for the selection of units which are much closer to the target linguistic contexts and therefore little or no modification of units is required. Festival's (Taylor et al., 1998, Black et al., 2001) Multisyn (Clark et al., 2004, 2007) is used in this thesis as the standard unit selection system. This system is used as the baseline unit selection synthesis system in the annual Blizzard Challenge (King and Karaiskos, 2011, 2012, 2013). The standard unit size is the diphone, however unit selection synthesis research has investigated many different unit sizes, such as phoneme-sized, half-phoneme-sized and frame-sized units (Conkie, 1999). It is worth noting that although diphone synthesis and unit selection synthesis may both use the diphone unit size, the differences in storage space and modifications performed mean these should not be confused with one another. Due to the numerous candidates from which to search for the best unit, unit selection requires a search to be performed. Viterbi search balances how well a candidate matches the target linguistic context, the target cost, with how well the selected units will join together, the join cost. The target cost typically relates to matching linguistic classifications between the candidate unit and the target unit, whereas the join cost typically relates to the acoustic properties of the units, reflecting how well two units will join together. The ratio of the target cost values relative to the join cost is usually trained by ear by experts in order to get the optimal unit selection synthesis

quality. Unit selection is able to exploit knowledge about neighbouring units from the unit database. Neighbouring units are given a join cost of zero, effectively meaning that variable unit sizes are present in synthesis. Once the unit sequence is selected for synthesis, overlap-and-add is performed to concatenate the units to create the synthetic waveform. In Multisyn no signal manipulation is performed, instead the system relies on the join cost to ensure there are no discontinuities. Other unit selection systems do perform signal manipulation at this point to ensure smoother speech, however such signal modifications risk introducing their own artefacts into the synthesised speech. Unit selection is still the cutting edge exemplar-based synthesis method, producing very high quality, natural-sounding speech, which is highly intelligible, when the required linguistic contexts are present in the training data and able to join together smoothly (Kaszczuk and Osowski, 2006, 2007, 2009, Chalamandaris et al., 2013, 2014). However there is a very wide range of possible linguistic contexts due to factors such as prosody and co-articulation with surrounding phonemes (Zen et al., 2009), meaning that it is extremely unlikely that the training data will cover the full range of sounds for the speech required to be output (i.e., zero target cost). Also, poor unit joins are extremely noticeable to the listener and drastically lower the quality and intelligibility of the speech. Exemplar-based speech synthesis systems benefit from ever-larger amounts of training data (Suendermann et al., 2010), as this is more likely to result in fewer joins in the synthesised speech being required. However the data must be from a single speaker, under similar recording conditions. Previous investigations have been made into bilingual unit selection synthesis (Esquerra et al., 1998).

1.5 Model-based synthesis

Model-based synthesis systems, referred to as statistical parametric speech synthesis (SPSS), produce the target speech at synthesis-time by learning models of speech from the training data, instead of storing the speech data directly (King, 2011). At the time work on this thesis began, the most common choice of method for constructing models of speech was to use decision tree-clustered Hidden Markov models (HMMs). In order to construct good models of speech, a parametrisation suitable for modelling speech is required. To do this, the speech waveform is typically passed through a vocoder. At synthesis-time, the decision tree is traversed by answering questions about the target linguistic context; at the leaf node of the decision tree is an HMM state which describes the distribution of vocoder parameters of the frames of speech whose linguistic con-

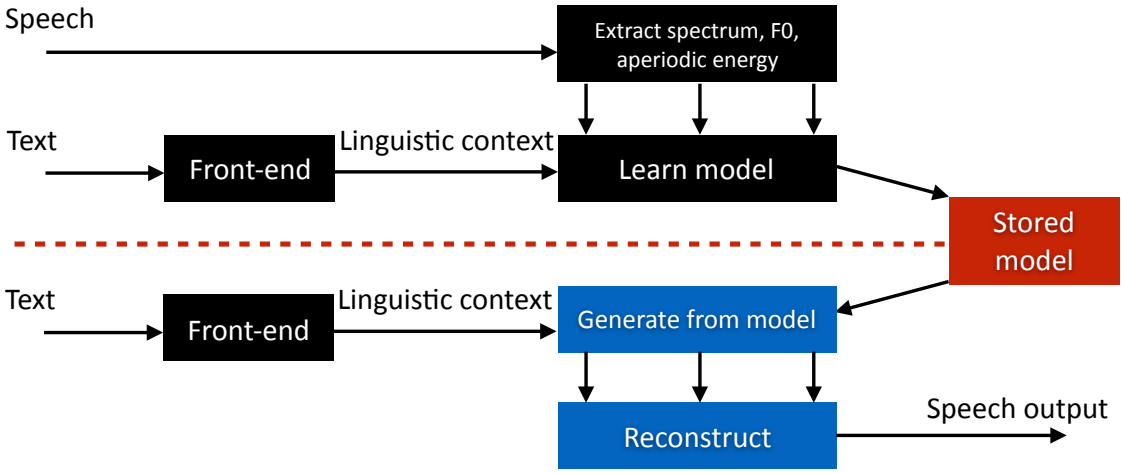


Figure 1.2: *Illustration of the statistical parametric speech synthesis pipeline. Above the red dotted line refers to the training phase and below the line refers to the synthesis phase.*

texts satisfy the decision tree traversal questions (Donovan and Woodland, 1995). Following the generation of speech parameters, these parameters are passed back through the vocoder to produce the speech waveform. The SPSS pipeline is shown in Figure 1.2.

SPSS systems are able to drastically reduce the footprint of required system resources from that of exemplar-based synthesis systems (Zen et al., 2009, Suendermann et al., 2010). This is due to only requiring storage space for the models of speech, instead of the waveforms of the entire training data. SPSS has been found to be able to function effectively with much less training data than exemplar-based systems (Yamagishi et al., 2008). This is because the models of speech are able to generalise across the speech examples observed in the training data in order to provide estimations of inevitable unseen linguistic contexts at synthesis-time (i.e., due to sparsity in the training data).

This method has been found to produce a consistent level of synthesis quality, over a wide variety of contexts and avoids the concatenation glitches produced by unit selection synthesis (Qian et al., 2010). However the quality level is below the best-case of the exemplar-based synthesis systems, i.e., the speech produced is not confusable for natural speech (King, 2011, Zen et al., 2009).

1.5.1 Vocoding

Vocoders fall into two categories: source-filter based and sinusoidal approaches. Source-filter vocoders represent the speech production process as a source excitation signal (vibrations from the glottis or frication) which is passed through a filter (the vocal tract). These vocoders therefore require good separation of source and filter contributions to speech (Valentini-Botinhao, 2013, page 17). Parametrisations of source and filter components of speech can be derived from source-filter vocoder output, which have been found to be modelled well (Tokuda et al., 1994, Koishida, 1998). Source-filter vocoders are the most common type of vocoders used for SPSS. These vocoders will be discussed in the remainder of this section. Sinusoidal vocoders represent speech as a combination of a periodic element (i.e., voiced component) and noise (Stylianou, 1996, Erro et al., 2007). Sinusoidal vocoders have been shown to provide very good quality speech coding, which is able to outperform source-filter models. However due to the large number of parameters, which may vary from frame-to-frame, they are more complex to model (Hu et al., 2014b), although significant research effort is being conducted to integrate these into SPSS (Banos et al., 2008, Erro et al., 2010, Hu et al., 2014a).

For source-filter vocoding, the complex task of separating source from filter – given only the speech waveform – is performed. The preferred method of separating these two components is to first calculate the spectral envelope (i.e., the vocal tract contribution). From this we can calculate the contribution of the source (i.e., the glottis) as the residual signal following inverse filtering (the remaining signal once the calculated spectral envelope has been removed). Rather than parametrising the residual signal, most vocoders used for SPSS, parametrise a representation of the excitation signal. Initially, simple vocoders were used which made a hard decision of using either a simple periodic pulse-train at the f_0 frequency or white-noise, to represent the excitation signal, for voiced and unvoiced sounds respectively (Yoshimura et al., 1999, Fukada et al., 1992, Imai, 1983). However this resulted in very low quality speech because this is an oversimplified representation of speech sounds, which are a mix of periodic and noise components. At the time work began on this thesis, STRAIGHT (Kawahara et al., 1999, 2001) was the standard vocoder in use for speech synthesis. STRAIGHT is a source-filter vocoder which uses mixed excitation, a mix of periodic pulse train and white noise for voiced sounds. The noise component is calculated by an aperiodicity spectrum. STRAIGHT applies smoothing of the spectrum in time

and frequency, during analysis, in order to provide a more stable calculation of the vocal filter and remove noise from harmonics of f_0. The smoothed spectrum, aperiodicity and f_0 are output by STRAIGHT, however these features are of a very high dimensionality and are highly correlated, therefore they require further processing in order to produce parameters which are more suitable for modelling. The vocal filter representation is further transformed to parameters suitable for modelling using the discrete cosine transform (DCT), the parameters are usually either Mel-cepstrum or line spectral frequencies (LSFs). The aperiodicity component is grouped into bands, band-aperiodicities (BAPs), to reduce the dimensionality, making the parametrisation more suitable for modelling. The f_0 component is transformed to log-f_0, this is because the log-scale is more perceptually meaningful. At synthesis-time, the generated parameters are transformed back to the vocoder parameters (spectrum, aperiodicity and f_0). From the vocoder parameters, a source is constructed, using mixed excitation, and passed through the vocal tract filter to produce the speech waveform.

In this thesis, the condition referred to as 'vocoded' speech is natural speech that has been passed through the vocoder and transformed to the *speech parameters used for modelling* before being transformed back to the time-domain waveform, rather than speech passed through the vocoder and transformed back to the time-domain waveform without any further parametrisation. This is an important distinction: the process of transforming speech to the parameters used for modelling results in further loss of quality, on top of the initial vocoding, due to this being a lossy process. This vocoded condition is included to provide a reasonable reference point in the investigations conducted in this thesis.

1.5.2 Hidden Markov model synthesis

Hidden Markov models (HMMs) are a popular method of modelling speech, being widely used in both speech recognition and speech synthesis. HTS (Zen et al., 2007a) is used in this thesis as the standard implementation of HMM synthesis. This system forms the baseline HMM condition for the annual Blizzard Challenge (King and Karaiskos, 2011, 2012, 2013). The description of HMMs in this thesis is based on that in Odell (1995).

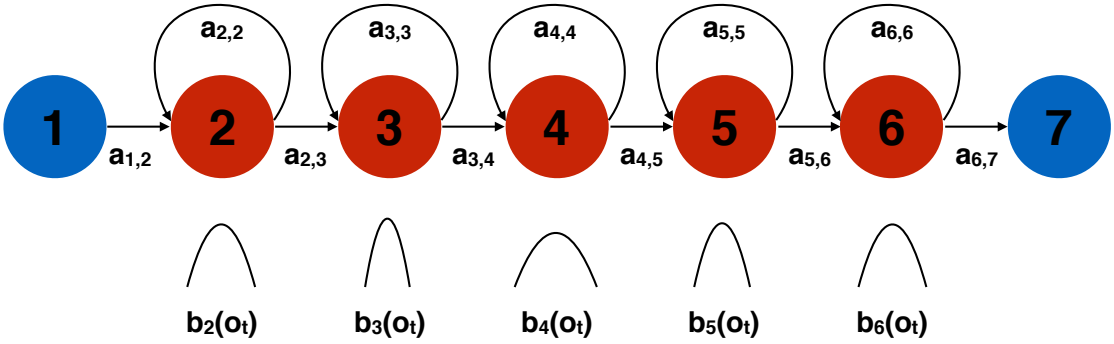


Figure 1.3: *Illustration of the standard 5 state left-to-right hidden Markov model topology. Blue indicates dummy state, red indicates emitting state. HTK-style state numbering is used.*

Training-time

HMMs are used to model speech parameters. As described in Section 1.5.1, for speech synthesis these speech parameters are derived from the spectrum output by the vocoder. For example an observed signal O, consists of T frame-level speech parameters (observations), $O = o_1, o_2, ..., o_T$. For simplicity let us say this is a recording of a phoneme. The HMMs generate the speech parameters in O with a sequence of states in our speech model, $X = x(1), ..., x(T)$. The exact sequence of states is unknown, hence 'hidden' Markov models. There is no memory of previous models at each point of time (the Markov assumption of conditional independence between observations) allowing for simpler computation. Although there are many possible model topologies (i.e., many ways in which we could transition between states) the standard for speech synthesis is to use models with five emitting states, with these states being in a left-to-right topology. This standard topology is shown in Figure 1.3: there are two dummy non-emitting states which are present to allow for models to be easily joined together, and HTK-style state numbering is used. The motivation for this topology is to capture the changes across time of the phoneme being modelled, as in speech this progression is very important, hence we force the model to progress from the first to the last state, through each emitting state.

Between each state is a transition probability $a_{i,j}$, which denotes the probability that the model in the current state i, will change to state j (for synthesis, an explicit duration model is used instead of the transition probability, this will be described later). Each emitting state has an associated function $b_i(\cdot)$, this is the probability that the

observation at time t was produced by the current state (i). In speech synthesis this is usually a single component Gaussian distribution. In practice, the speech signal relating to an utterance contains a sequence of phonemes. In order to model this, the phoneme models corresponding to the phoneme state sequence in the utterance are concatenated together to produce a model for the utterance.

At training-time, models are created by first calculating a global mean and covariance estimation (assuming diagonal covariance) of the speech parameters, across the training data. The global mean and covariance estimation is then used to initialise context-independent monophone models. The model of the utterance is constructed by joining the monophone models corresponding to the phone sequence in the training utterance: the dummy start and end states of the HMMs make this a simple process. The state-wise monophone models are then updated to maximise the likelihood of the observations which are generated by each state, using Baum-Welch re-estimation. The Baum-Welch algorithm, also known as the expectation maximisation (EM) algorithm, calculates $\gamma_j(t)$, the probability that observation o_t was generated by state j. Baum-Welch makes use of two recursive procedures, the forward pass and the backward pass. The forward pass calculates, $\alpha_j(t)$, the probability that the observations from time 1 to time t, result in being in state j at time t. The backward pass then calculates, $\beta_i(T)$, the probability of the remaining observations in the signal[1]. The HMMs are then updated as described in equations 2.25 and 2.26 in Odell (1995), page 18. The resulting monophone models are then cloned to produce context-dependant HMMs.

The context-dependant HMMs are then clustered using decision tree regression. Separate decision trees are built for each individual parameter stream (Mel-cepstra, log-f_0 and BAP) and for each of the states within these parameter streams. Decision tree regression can be seen as dividing up the 'space of all speech sounds'. The aim is to divide the 'space' up such that each of the clusters of speech sounds created following this step is sufficiently large to generalise for linguistic contexts which are unseen in the training data. At the same time the division of the 'speech space' is sufficient such that the subsequent models of speech at each of these clusters is pure enough, i.e., the models will not be distorted by mismatching linguistic contexts. At the root node, the HMMs are all combined to provide a maximally-likely representation of the data given one model. Estimates are then made as to what the gain in likelihood would be from switching from having the current one HMM to represent the data, to splitting to two HMMs according to the answer to a question about linguistic context. The answers

[1] Calculations of probabilities use logarithms to avoid numerical underflow.

to these question are binary (yes/no). The aim is to maximise the likelihood of the parameter sequence, given the model. As we have computed context-dependant HMMs we are able to estimate what the new likelihood would be given these different hypothetical splits in the decision tree without actually having to keep retraining the models. The split in the linguistic contexts that maximises the likelihood is selected as the split in the decision tree. Splitting of the nodes is performed until a stopping condition is met, this stopping criterion is typically minimum description length (MDL) (Shinoda and Watanabe, 2000) or when the number of linguistic contexts within the node in the decision tree, is below a minimum threshold. Splitting the linguistic contexts will always result in an increase in likelihood, however splitting data too much means that the models won't be able to sufficiently generalise for unseen linguistic contexts. MDL balances the gain in likelihood from performing a split in a node in the decision tree against how many parameters (i.e., questions in the decision tree) are used, if the split does not sufficiently increase the likelihood then it is decided that the split shouldn't be made and the node is sufficiently clustered. This method of performing splits in the decision tree assumes that the alignment of the models to the training data does not change as a result of the different possible splits in linguistic contexts. As this is not strictly true, following the construction of the decision tree, the whole decision tree process can be repeated to further improve the models. Each state, within each of the parameter streams, is clustered using a separate decision tree. An example of decision tree regression is shown in Figure 1.4. Multi-space probability distribution (MSD) HMMs (Tokuda et al., 2002) are used to model log-f_0, as this parameter stream is not strictly continuous due to f_0 being undefined for unvoiced regions of speech. Log-f_0 is therefore modelled in two spaces: voiced and unvoiced speech.

In speech recognition, and therefore at training-time for speech synthesis, using transition probabilities to judge appropriate phoneme duration performs adequately (Odell, 1995, pages 12–13). However at synthesis-time, using the HMM transition probabilities to predict durations has been found to dramatically reduce synthesis performance. As a result, state duration is modelled by a separate decision-tree clustered Gaussian distribution, which is used to predict durations at synthesis-time (Yoshimura et al., 1998). As the duration of each state is determined by the duration Gaussian and not by the current HMM state, the synthesis configuration is not strictly a HMM, therefore the configuration is referred to as hidden semi-Markov models (HSMMs) (Zen et al., 2004, 2007b).

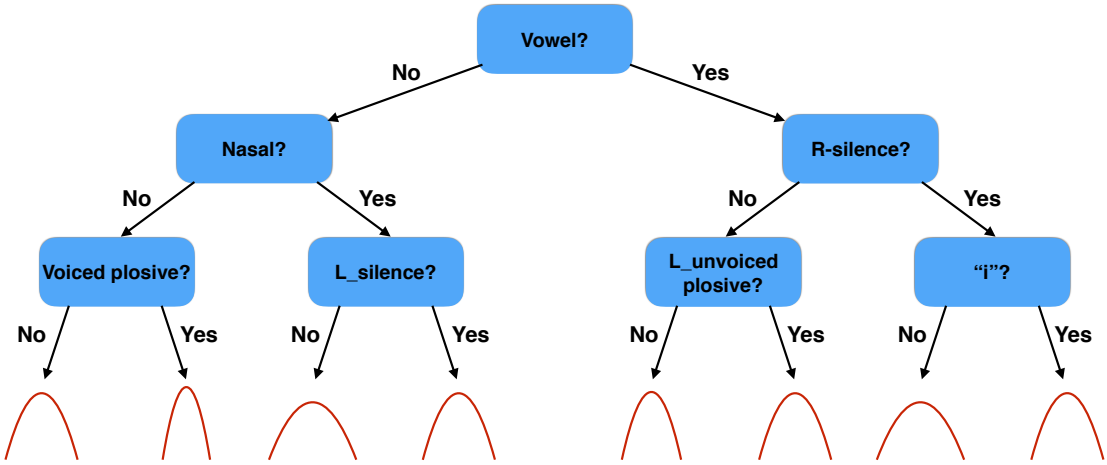


Figure 1.4: *Illustration of decision tree regression of HMMs. Based on Figure 2.14 in Yamagishi (2006).*

Synthesis-time

At synthesis-time, the decision trees are traversed by answering the questions about linguistic context used to split the parameter space. Once a leaf node is reached, the Gaussian distribution associated with it is used for synthesis. Once the HMMs are selected there is the problem of how to use the associated parameter distributions to synthesise speech. This is a problem because if we were to take the most likely parameter for each frame (i.e., the mean of the current model), this would result in a flat trajectory which steps suddenly at every state boundary (as shown in Figure 1.5). This parameter trajectory results in poor quality synthesised speech. What is required is to generate parameter trajectories which smoothly vary across time, as is the case for natural vocoded parameters. In order to achieve this, delta and delta-delta parameter attributes are added to the speech parameters which are used for training. Following the addition of these attributes, it is possible to factor in more information about how the parameter trajectory should vary across time. Generation of parameters, considering such variance across time, is done using maximum likelihood parameter generation (MLPG) (Tokuda et al., 2000, Gales and Young, 2008, Watts, 2012). The description of MLPG provided here is based on that of Gales and Young (2008) and Watts (2012). In MLPG, the generated frame-level static parameter values are calculated using;

$$\mu_Q^s = (W^T \Sigma_Q^{-1} W)^{-1} W^T \Sigma_Q^{-1} \mu_Q \tag{1.1}$$

Where W contains the relationship between frame-level static parameters which are to be factored into the generated parameter trajectory (i.e., static, delta and delta-delta relations (Yamagishi, 2006)). For example the delta relationship between neighbouring static frame values (o_t^s) can be found for a single frame by the matrix W^s;

$$W^s = \begin{bmatrix} 0 & 1 & 0 \\ -1 & 0 & 1 \end{bmatrix} \tag{1.2}$$

$$o_t = \begin{bmatrix} o_t^s \\ \Delta o_t^s \end{bmatrix} = W^s \begin{bmatrix} o_{t-1}^s \\ o_t^s \\ o_{t+1}^s \end{bmatrix} \tag{1.3}$$

To include delta-delta information as well as delta information the matrix W^s can be expanded to:

$$W^s = \begin{bmatrix} 0 & 0 & 1 & 0 & 0 \\ 0 & -1 & 0 & 1 & 0 \\ 1 & 0 & -2 & 0 & 1 \end{bmatrix} \tag{1.4}$$

$$o_t = \begin{bmatrix} o_t^s \\ \Delta o_t^s \\ \Delta^2 o_t^s \end{bmatrix} = W^s \begin{bmatrix} o_{t-2}^s \\ o_{t-1}^s \\ o_t^s \\ o_{t+1}^s \\ o_{t+2}^s \end{bmatrix} \tag{1.5}$$

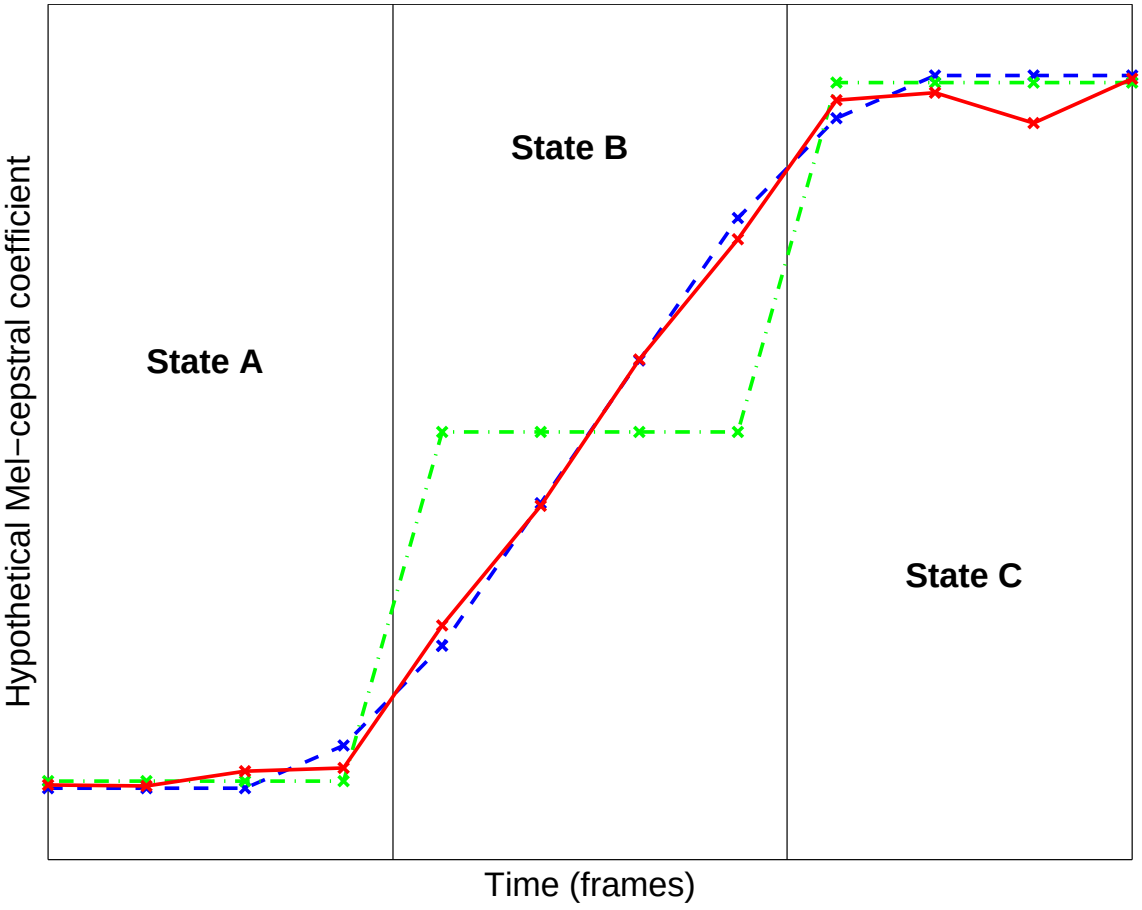


Figure 1.5: *Demonstration of MLPG. The dashed blue line shows the natural trajectory, the dashed green line shows the naïve step-wise realisation of the state-wise mean, the solid red line shows the trajectory generated by MLPG, using delta and delta-delta features. Illustration based on the example in Watts (2012), page 23.*

By repeating the W^s matrix, with a shift for each frame and padded with zeros, the utterance-level W can be derived. W is an RT by T matrix, where R is equal to the number of acceleration features (static, delta and delta-delta) and T is the number of frames in the utterance. See Watts (2012), page 22 for an example W matrix. The frame-level distributions returned by the HMM for the utterance to be synthesised (μ_Q and Σ_Q, represent the means and covariances respectively) can be used alongside the relation matrix W, to generate the parameter trajectory (μ^s_Q) which considers how the parameters should vary over time, by using equation 1.1. Figure 1.5 shows a comparison between using a naïve step-wise realisation of the state-wise mean and using MLPG to generate a hypothetical parameter trajectory.

1.6 Evaluation of speech synthesis

Synthesis evaluation is in itself a subject of research. Here I will discuss the methods of evaluation used in this thesis, however there exist many other possible approaches to evaluation. For example, intelligibility testing was not performed in this thesis. This is because the primary focus of the thesis is on improving the quality of SPSS, given that SPSS is typically highly intelligible but the quality of speech is what is currently holding this synthesis domain back, as it is not confusable for natural speech.

The aim of speech, and therefore also speech synthesis, is to convey information to the listener and do so in a way which is pleasant and engaging for the listener. Because of this, subjective evaluation of synthesis systems will always be of paramount importance, as human listeners are the target audience of such systems. As such, subjective testing is the primary form of testing performed throughout this thesis.

1.6.1 Mean opinion score

One of the most common ways to evaluate speech synthesis is via a mean opinion score (MOS) test. MOS testing is an example of an opinion score task, where listeners are asked to rate a given sample of speech. This is an established standard method of subjective testing within speech synthesis and is performed within the Blizzard Challenge (King and Karaiskos, 2009, 2010, 2011, 2012, 2013). MOS tests usually present one speech example at a time to the listener, i.e., the listener judges one utterance under one of the conditions to be tested in isolation. Listeners are then required to rate the quality or naturalness of the speech on a scale from 1 to 5, with 1 referring to ‘bad’ and 5 referring to ‘excellent’. It is common practice to perform careful balancing of speech samples used for MOS testing to remove potential effects of repeated listening of the same utterances by the participant. Listening tests need to be balanced by design in order to ensure that ratings of different conditions on the same sentences are present, to remove sentence-specific judgements from listener responses, whilst also removing effects of listening to the same sentence multiple times. A typical approach to removing this issue is to use a balancing method such as a latin square test design (MacKenzie, 2013, pages 177–181).

1.6.2 MUSHRA

MUSHRA (MUltiple Stimuli with Hidden Reference and Anchor) (ITU Recommendation ITU-R BS.1534-1, 2003) is an established testing paradigm in the field of speech coding and is just beginning to be used within speech synthesis (Henter et al., 2014a). MUSHRA presents a single sentence to the listener under all conditions on a single screen, thus allowing the listeners to make their judgements with knowledge of the full range of conditions. In the MUSHRA tests performed in this thesis, listeners were asked to rate the speech on a scale from 0 (completely unnatural) to 100 (completely natural). The test includes a hidden reference (this is usually natural speech) which acts as an upper-bound from which listeners can make their judgements. Listeners are informed that the upper anchor condition is hidden among the range of conditions included in the test and are asked to rate this as 100, with the other conditions rated accordingly. When MUSHRA is used within the field of speech coding, it usually also includes a lower anchor (e.g., low-pass filtered speech) so as to stabilise listeners responses. However, within the field of speech synthesis, the task of identifying a suitable lower anchor is an extremely complex one. Therefore there is no lower anchor used in the MUSHRA testing within this thesis.

This paradigm is primarily an opinion score test. However, as the listener is making judgements across all conditions at once, MUSHRA also allows for interpretation of listener responses based on the rank order assigned to conditions, ignoring the absolute values of their scores. The ability to perform multiple analysis from one listening test makes MUSHRA testing quite powerful. However MUSHRA testing is more time-consuming than MOS testing, with listeners being required to cross-verify their ratings for each set of ratings (one screen of the same sentence under the full range of conditions included in testing). This must therefore be considered when deciding on which form of subjective testing to use. Ribeiro et al. (2015) ran the same listening test using a MUSHRA and a MOS test and noted the added benefit of side-by-side rating in the MUSHRA paradigm. However Ribeiro et al. also noted, informally, that there was a noticeable difference in the time taken by participants for these two different listening tests.

1.6.3 'Same or different' task

A 'same or different' task is where two samples of speech are played to the listener, and the listener is simply asked if the two samples are the same or different in terms of the criterion used for testing. For example if the test is being conducted with respect to synthesis quality, listeners are asked to judge whether the two samples of speech are the same or different in terms of the synthesis quality. This approach of testing is used extensively in Part I of the thesis because it doesn't require guidance of the listener to attend to a particular aspect of the speech and instead the listener is left to make this binary choice without bias from the experimenter. The listener responses to the "same or different quality" task can then be pooled together to provide a perceptual distance matrix between the conditions tested. This matrix can then be processed to better understand the properties of speech attended to by the listeners. In this thesis, the perceptual distance matrix is processed using multidimensional scaling (MDS), in order to project the perceptual distances into a space with a number of dimensions defined by the user. The MDS space can then be visualised for interpretation.

1.6.3.1 Multidimensional scaling

MDS is an effective tool for interpreting listener responses as it allows us to visualise the complex inter-condition relations (i.e., how close each of the conditions are to each other) in a manner which is more simple to interpret than a box-plot or bar chart. This method of testing and analysis of results is used in this thesis as it has been found to be very illuminating when teasing apart the perceptual differences between speech stimuli (Mayo et al., 2005, 2011). MDS analysis is especially effective when it is suspected that listeners are using more than one perceptual dimension to make their judgements (something that a perceptual test such as MOS cannot discover).

Ordinal (or non-metric) MDS using Kruskal's Stress-1 criterion is used in this thesis (Borg et al., 2013). The stress value (i.e., the value returned by the Stress-1 criterion) is an indication of the degree to which the current projection is an accurate representation of the data. Ordinal MDS means that data points (i.e., different conditions included in the perceptual test) are viewed in terms of rank order instead of using absolute distance values. The description of MDS in this thesis is based on that in Borg et al. (2013).

Points are initialised in the MDS space using classical MDS, which performs the transformation of the data assuming the dissimilarity measure in the matrix input to

the MDS procedure represent Euclidean distances. The perceptual matrix which we are performing MDS on is not a matrix of Euclidean distances; this step is only performed to provide a starting point from which iterative updates to the MDS space can be performed. Ordinal MDS uses an iterative procedure to update the placement of the conditions within the MDS space in order to optimise the Stress-1 criterion. The Stress-1 criterion is defined as follows:

$$\textit{Stress-1} = \sqrt{\sum_{i<j}(d_{ij}(X) - \widehat{d}_{ij}))^2 / \sum_{i<j}(d_{ij}^2(X)} \tag{1.6}$$

Where X denotes a configuration of points in the MDS space, $d_{ij}(\cdot)$ is the distance between conditions i and j in the MDS space and $\widehat{d}_{ij}$ is defined as:

$$\widehat{d}_{ij} = f(p_{ij}) \tag{1.7}$$

Where p_{ij} is the proximity between conditions i and j from the data. Disparities ($\widehat{d}_{ij}$) are computed using a monotone step function regression (f) of the proximities onto the distances. Iterative updating of the points in the MDS space is performed by taking turns to; move points in the MDS space, and update the disparities. This is done as follows:

1. The disparities are frozen; the points in the MDS space (X) are moved so that the distances minimise the Stress-1 criterion.

2. The points in MDS space are frozen; the disparities are re-scaled so that the Stress-1 criterion is minimised.

This iterative updating of the MDS space is repeated 200 times or until the stress value is not reduced by more than 0.0001. The Matlab implementation of MDS based on Kruskal's normalised Stress-1 criterion[2] was used to provide the MDS visualisations generated in this thesis (Kruskal and Wish, 1978). One point worth noting is that the iterative procedure used for MDS does not guarantee the globally optimal solution will be found.

MDS requires a trade-off between representing the data accurately and representing the data in a dimensionality which is low enough to allow it to be visualised and

[2]Function 'mdscale' from the Matlab statistics toolbox.

interpreted. Obviously the stress level will always continue to decrease as the number of dimensions used is increased, however the practicality of visualising and interpreting the data must also be considered. The preference would be to always plot the data in two dimensions, as this allows for the simplest visualisation of the data. However, this obviously depends on the suitability of the data to being represented in two dimensions. In this thesis, two dimensions are used for MDS plots where there's an obvious 'elbow' in the stress values at the point of fitting the data to two dimensions (i.e., the gradient of the decline in the stress at differing numbers of dimensions dramatically drops off after two dimensions). Where there is no obvious elbow in the stress values, three dimensions are used to represent the data. This is because three dimensions is the largest number of dimensions which is still feasible for interpretation.

1.7 Improvements to basic systems

So far in this chapter we have discussed the standard HMM synthesis system configuration. However, this standard configuration produces speech which is not confusable for natural speech. In response to the observed 'ceiling' effect of achievable naturalness from SPSS systems, a number of improvements to the standard HMM synthesis system have been proposed in the literature. A number of these will now be discussed.

1.7.1 Effect of vocoding

Vocoding has long been identified as a cause of reduced quality of speech (Zen et al., 2009). Despite the increase in the performance of vocoding, following the introduction of the STRAIGHT vocoder, there are still many claims in the literature citing the vocoder as a limiting factor in speech synthesis quality (Zen et al., 2009) and many researchers trying to overcome these issues.

1.7.1.1 Alternative vocoders

In order to improve the quality of vocoding, alternative source-filter-based vocoders have been developed. As discussed in Section 1.5.1, the benefit of source-filter vocoders is that parametrisations are available which are established as being easily modelled (e.g., Mel-cepstra). Some researchers have suggested that the parametrisation of the voice source (i.e., the glottis) by existing vocoders, such as STRAIGHT, reduces the quality of the resulting speech. A number of vocoders overcome this by using a glottal

pulse (i.e., the residual) from the speech waveform to represent the source component, instead of using a more simple pulse train mixed with noise. The GlottHMM vocoder (Raitio et al., 2011b) constructs a library of glottal pulses at training-time. At synthesis-time glottal pulses are selected from the library of pulses. Alternatively, in Cabral (2011) there is an explicit model to produce the glottal flow which is to be used for synthesis. Another alternative is suggested by Drugman and Dutoit (2012), where principal component analysis (PCA) is performed on the residual signal. For the vocoders discussed in this section, the glottal pulse is represented up to a maximum voiced frequency, above this frequency noise is used.

1.7.1.2 Hybrid approaches

An alternative to improving the vocoder is to remove the use of vocoding from synthesis altogether and instead use SPSS to guide the selection of units for unit selection synthesis. Hybrid approaches to speech synthesis aim to combine the benefits of exemplar-based synthesis and model-based synthesis. The aim is therefore to produce synthesised speech which has the extremely high quality of unit selection whilst also having the flexibility and consistency in the quality of synthesised speech of SPSS. This synthesis domain will be investigated in Part II of the thesis.

Hybrid synthesis can take a number of forms. For example, in Black and Taylor (1997), automatic clustering of units (i.e., preselection) is performed in order to improve the selection of units in unit selection synthesis. Commonly, hybrid systems use either SPSS-generated parameters (Qian et al., 2010, 2013) or the Gaussian distributions associated with SPSS models (Yan et al., 2010) to guide unit selection. These systems have been investigated in Ling and Wang (2006, 2007, 2008), Xia et al. (2014), Hirai et al. (2007), and have been used for both target and join cost calculations in unit selection synthesis. Many of the system variants investigated are similar to the system variances performed in unit selection synthesis systems in general, for example changes in the unit size to be used. In Ling and Wang (2008), the training criterion of the SPSS system was adapted to make it more suitable for performing unit selection.

Hybrid synthesis can be extended further by combining waveforms and generated parameters, referred to as multi-form synthesis (Pollet and Breen, 2008, Sorin et al., 2011, 2012, 2014, Fernandez et al., 2015). In multi-form synthesis, the decision whether to use modelled or real speech examples can be made using a measure of which linguistic contexts will be generated well by SPSS. For example, highly stationary sounds are predicted well by SPSS (Sorin et al., 2012) therefore for such linguistic

contexts the speech parameters predicted by the SPSS system can be used in the output synthesis. By using such a metric, the number of units which are required to be stored can be reduced allowing the footprint of the synthesis system (a shortcoming of unit selection synthesis) to be reduced. Multi-form synthesis can be exploited by using real examples of speech where there is a large enough number of contiguous units suitable for selection, resulting in best-case unit selection synthesis, due to a low number of joins being present. However where this isn't possible the parameters generated by the SPSS system can be used instead to eliminate the presence of numerous joins in the synthesis (Fernandez et al., 2015).

Hybrid speech synthesis systems have been found to be effective at combining the benefits of both the underlying unit selection synthesis system and the SPSS system used to drive the selection of units. For example hybrid synthesis systems have performed very well in Blizzard Challenges (Chen et al., 2011, Ling et al., 2012, Chen et al., 2013).

1.7.2 Quality of training criteria

Black identified the state-wise predictions of standard HMM synthesis as a possible limiting factor and constructed the decision tree in a frame-wise manner, instead of being strictly state-wise, in the clustergen system (Black, 2006). The system uses identifiers for each frame within a state and includes questions about progress through the state for decision tree clustering. This therefore means that the decision tree may produce splits in the linguistic contexts relating to the current frame. At synthesis time, each frame-level distribution is selected by traversing the decision tree.

Yan et al. identified decision tree clustering of differing linguistic contexts as having a detrimental effect on quality. Yan et al. (2009) introduces rich-context HMM synthesis, where distributions from a standard decision tree clustered HMM system are used as a target from which to select rich-context HMMs. The means of the distributions associated with rich-context HMMs are trained only on frames of speech where the linguistic contexts match exactly and therefore no averaging across differing linguistic contexts is present. Rich-context HMM synthesis is investigated in Part II of the thesis.

At the time work began on this thesis, Deep Neural Networks (DNNs) had not yet re-emerged as a method for speech synthesis, therefore the focus of the thesis is on HMM speech synthesis. The use of DNNs for speech synthesis will be discussed in Chapter 8.

1.7.3 Post-generation approaches

Another approach to overcoming the shortcomings of SPSS is to generate speech parameters using SPSS, as normal, and then apply fixes to the parameters after generation. These post-generation fixes aim to undo the effects of modelling by exploiting knowledge of shortcomings of generated trajectories from SPSS and altering the parameters accordingly so they more closely resemble natural speech.

1.7.3.1 Postfiltering

Although MLPG generates parameter trajectories which vary smoothly across time, these trajectories often have a reduced variance, i.e., peaks and valleys in the trajectory are less pronounced. This is due to the generation procedure aiming to produce the most likely trajectory, resulting in the generated parameters sitting very close to the mean of the state-level distributions used across the utterance. Given the reduced utterance-level (i.e., global) variance in SPSS-generated parameter trajectories, different techniques have been devised to restore more natural variance. Postfiltering was a very early post-generation technique to improve the global variance of a generated trajectory (Zen et al., 2009). Postfiltering, as described in Koishida et al. (1995), scales the parameters above the first Mel-cepstrum coefficient by β. This is effectively:

$$\bar{c}_t(m) = \begin{cases} (1+\beta)c_t(m), & \text{if m} > 1 \\ c_t(m), & \text{otherwise} \end{cases} \tag{1.8}$$

Where m is the Mel-cepstra coefficient number, $c_t(m)$ is the generated Mel-cepstra parameter at time t and $\bar{c}_t(m)$ is the parameter after postfiltering.

1.7.3.2 Global variance

In order to better reinstate natural global variance into the SPSS-generated trajectory, than was the case with postfiltering, the idea of constructing a global variance (GV) model was devised. To do this, the average variance across each utterance in the training data, per parameter, is calculated. A Gaussian distribution is then fitted to these variance values.

At synthesis-time, the parameter trajectory generated by MLPG is then weighted against satisfying the GV distribution. The derivation of the trade-off between producing parameters which satisfy the maximum likelihood and which satisfy the GV distribution, can be found in Toda and Tokuda (2007). Context-dependent GV models can be used to model GV more precisely.

Alternatively, Silén and Helander found that variance scaling can perform almost as well as GV, whilst being computationally much cheaper. In Silén and Helander (2012) the average global variance of each natural vocoded parameter coefficient is calculated across the training data, $(\sigma_m^{gv})^2$ where m denotes the coefficient number. At synthesis-time, the global variance of the trajectory generated by MLPG is improved by:

$$c_m^{'}(t) = \frac{\sigma_m^{gv}}{\sigma_m}[c_m(t) - \mu_m] + \mu_m \tag{1.9}$$

where t denotes the current frame, μ_m and $(\sigma_m)^2$ denote the utterance-level mean and variance of the coefficient m. This form of variance scaling is used in Part I of the thesis.

1.7.3.3 Modulation spectrum

Modulation spectrum postfiltering aims to restore temporal detail lost during the process of modelling and generating parameters from the models. The modulation spectrum is a measure of the parameter trajectory as a power spectrum, i.e., how the trajectory varies across time, calculated using the discrete Fourier transform (DFT). The modulation spectrum across the natural vocoded parameters of the training data is calculated and to each modulation frequency and parameter coefficient pairing a Gaussian distribution with diagonal covariance is fitted. The utterances in the training data are then generated by SPSS (using MLPG only or considering GV as described in Section 1.7.3.2) and the parameter trajectories are transferred into the modulation spectrum domain. Again a Gaussian distribution with diagonal covariance is fitted to each modulation frequency and parameter coefficient pairing. The natural and SPSS modulation spectrum Gaussian distributions are then stored for use at synthesis-time.

At synthesis-time, generated parameter trajectories are transferred into the modulation spectrum and scaled from the SPSS modulation spectrum distribution towards the distribution of natural vocoded speech. The strength of the filter, i.e., how far parameters are scaled from SPSS towards natural speech is controlled by a postfilter emphasis coefficient (Takamichi et al., 2014b). This form of postfiltering has been found to improve the quality of synthesised speech (Takamichi et al., 2014a, 2015). Modulation spectrum postfiltering is investigated in Part I of the thesis.

1.7.4 Summary

Despite several years of improvements, the quality of synthesised speech from SPSS remains significantly less natural than speech output from unit selection synthesis systems under 'best-case' conditions (King, 2011, Zen et al., 2009) and natural speech. This is consistently reflected in the results from the Blizzard Challenge (King and Karaiskos, 2009, 2010, 2011, 2012, 2013), even though much progress has been made (King, 2014, Tao et al., 2014).

The cause of this reduced quality is commonly attributed in the literature to "over-smoothing" (Takamichi et al., 2013, Hojo et al., 2013), and that this is the fault of the statistical model. However, to the best of my knowledge, there are no formal, published studies supporting this claim. The idea of "over-smoothing" is at first glance seemingly a simple one, but may in fact combine a number of different effects from the signal representation and statistical modelling in use, both in the spectral and temporal domains. Smoothing is inherent in the statistical modelling framework, of course. The spectral envelope is smoothed first by the low-dimensional representation, then again by averaging over consecutive frames and over multiple tokens. The temporal structure of the speech parameters is smoothed because the model represents the trajectory with limited resolution (e.g., 5 states per phone-sized-unit). In this thesis the term 'smooth' will refer to temporal smoothness, i.e., the absence of noise in the parameter trajectory. This thesis will use the term 'reduced variance' to refer to parameter trajectories with less accentuated peaks and valleys. These two properties have been clearly defined in order for their respective contributions to the quality of synthesised speech to be better understood.

1.8 The problem of the current research methodology in the pursuit of synthesis improvements

Typically, in the literature the reduced quality of speech synthesised from SPSS systems is attributed to a hypothesised cause, seemingly without confirmation from formal studies, and improvements are made to the SPSS system to overcome this. However, due to the lack of formal investigation before trying to address the cause of reduced quality, there is every chance that the proposed improvement may not result in significant improvements. Even if improvements are found, it is unknown whether a large underlying issue has been fixed rather than a more minor one. This method of research is inefficient and may result in wasted effort in finding large-scale improvements. Therefore, in this thesis, I formally investigate the effect of hypothesised causes of reduced quality in SPSS. This investigation is similar to the investigation undertaken in Morgan et al. (2013), which investigated shortcomings in HMM speech recognition. The investigations in this thesis are conducted to provide a check list of elements which degrade the quality of SPSS systems, alongside an understanding of how much each element degrades the quality. The findings from the investigations will then be used to motivate improvements to SPSS.

In order to more effectively improve the quality of SPSS, a framework which can separate out the different contributions of the various processes of modelling is required. This is the contribution of Part I of this thesis. The findings of this investigation will then be used to apply informed improvements to SPSS in Part II of the thesis, which fixes the highlighted issues.

Part I

Investigating the shortcomings of HMM synthesis

Chapter 2

A methodology for investigating the shortcomings of HMM synthesis

2.1 'Continuum' of speech between natural and modelled speech

Section 1.8 highlighted a problem with the current research approach in speech synthesis. Hypothesised causes of reduced quality within HMM synthesis systems are used to inform the future research direction without formal testing to confirm the hypothesis. This approach to conducting research results in a lack of knowledge as to whether proposed solutions will really fix speech synthesis problems. Also if improvements are found, it is unknown whether a large underlying issue has been fixed rather than a more minor issue. Such an approach to research is obviously sub-optimal and may lead to research effort either being focused in resolving a minor issue with synthesis rather than a much more major underlying issue or in attempting to fix incorrectly hypothesised causes.

Instead this thesis aims to formally test a large range of hypothesised causes of reduced synthesis quality over a collection of investigations in order to determine where attention in future research should be focused. The perceptual findings from these investigations will provide insight into the underlying causes of reduced quality from which informed improvements to HMM speech synthesis will be implemented.

In order to separate out hypothesised causes of reduced synthesis quality, we will view each hypothesised cause as an element within a 'continuum' of speech. This continuum goes from natural speech at one end through to modelled speech at the

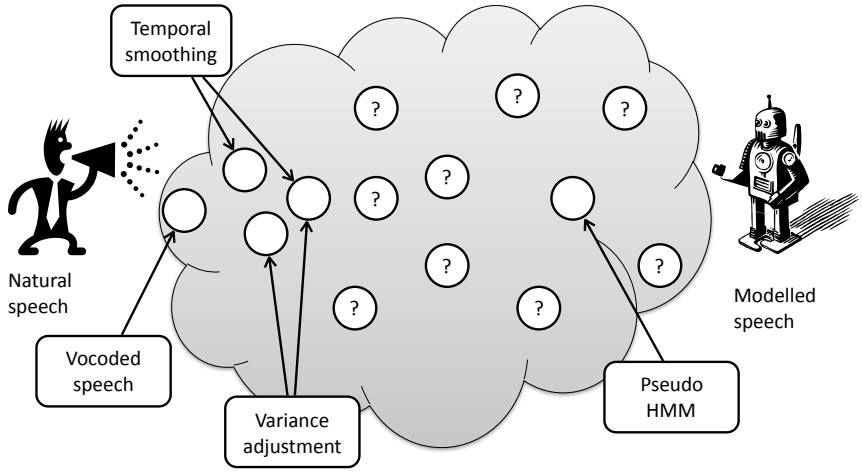


Figure 2.1: *Illustration of concept of modelled speech being the result of a series of effects applied to natural speech. Here a few example points along this continuum are given.*

other, as illustrated in Figure 2.1. Any or all of the elements within the continuum may introduce their own artefacts which have a detrimental effect on the quality of speech that is achievable at synthesis-time. The various effects and assumptions hypothesised to be causes of reduced synthesis quality co-occur in a full HMM speech synthesis system making it difficult to know their contribution to subsequent synthesis quality. However by investigating these as individual elements along a continuum we can piece apart different hypothesised causes and measure their effect independently.

Using this continuum concept we can perceptually test speech following each of the hypothesised causes of reduced naturalness. These perceptual test findings will be able to confirm or reject hypothesised causes of reduced quality of synthesis from the literature. Each cause can also be quantified in order to determine where large gains in the quality of speech synthesis can be made.

2.2 Methodology of HMM simulation framework

We can investigate a range of different effects and assumptions present within statistical parametric speech synthesis systems by adopting a continuum of hypothesised causes of reduced naturalness. This allows us to test hypothesised causes which are commonly attributed with reducing quality of output speech but have not until now

been formally tested. In no particular order, these include:

- The effect of temporal smoothing (investigated in Chapters 3, 4 & 5).
- The effect of global variance (investigated in Chapters 3, 4 & 5).
- The effect of averaging across differing linguistic contexts (investigated in Chapter 4).
- The effect of independent modelling of parameter streams (investigated in Chapters 5 & 6).
- The effectiveness of covariance modelling (investigated in Chapter 6).
- The effectiveness of current enhancement methods on modelled speech (investigated in Chapter 5).
- The effect of vocoding (investigated in Chapters 4, 5 & 6).
- The effect on different parametrisations of speech (investigated in Chapter 4).

This investigation of the causes of reduced quality in HMM synthesis can be conducted in a variety of ways. In this thesis, two different approaches are presented. The first of these is to 'simulate' different effects of modelling (i.e., manipulate speech parameters to follow documented characteristics of modelled speech which is hypothesised in literature as reducing synthesis quality) and also apply oracle knowledge to the models of speech parameters used. By simulating hypothesised causes as elements in the continuum between natural and standard HMM modelled speech, perceptual testing of the various elements gives an insight into the detrimental effect introduced by each of these. These results will therefore indicate which hypothesised causes are genuine, and will identify which have the largest delimiting effect on the quality of HMM synthesis. This framework of investigation is conducted in Chapters 3, 4 & 5. Secondly, we can also investigate the detrimental consequences of different assumptions which are applied in the HMM synthesis paradigm. To do this however requires a specially crafted corpus of repeated speech. With such a corpus, independence assumptions which are standard within statistical parametric speech synthesis systems can be applied using 'perfectly natural' trajectories from recordings of the same sentence. The resulting speech will therefore observe the upper-bound performance with certain

modelling assumptions in place given that ‘perfectly natural’ generation of the subsequent speech parameters is possible. This investigation will be explained in Chapter 6.

Chapter 3

Initial implementation of investigation methodology

This chapter is an expanded version of the work in Merritt and King (2013) and therefore the text is closely related to that.

3.1 Implementing the framework

The framework for simulating and perceptually testing separate effects of modelling, as described in Chapter 2, will now be introduced. To demonstrate its use a couple of the potential causes of the degradation in naturalness introduced by the use of statistical models will be tested. The framework is general and could be applied to many different aspects of the problem. Chapters 4 & 5 extend this methodology further. The idea is to *simulate* the effects of modelling vocoded speech parameters, in a carefully controlled manner. Following this, perceptual tests are then conducted in order to observe how these conditions individually affect the naturalness of the generated speech. Knowledge obtained by such experiments can then be used to identify aspects of statistical parametric speech synthesis systems which are causing the largest degradations. From such findings, improvements to the synthesis system can then be developed.

Current HMM-based synthesisers are large, complex systems. There are interactions between the signal processing (e.g., how the spectral envelope is extracted and how it is represented for the purposes of modelling) and the modelling (e.g., the parameter sharing structure of the model and how much data is available to estimate each free parameter) which need to be investigated. In the work presented in this chapter, this will be done by removing the modelling part completely and replacing it with

a series of operations which are designed to simulate some modelling effects. The proposed approach allows us to vary the strength of these effects, and to examine the interactions between them. Thus, by using simulations, it is possible to continuously vary the system from being a simple vocoder at one end of the scale, to a simulated HMM synthesiser at the other. In this chapter, the effects used are temporal smoothing and variance scaling of the speech parameters representing the spectral envelope.

3.2 Measuring the effects

The second component of the proposed framework is perceptual testing of the resulting quality of speech following the simulated effects of statistical modelling. Asking listeners to attend to specific aspects of the speech is problematic (Mayo et al., 2011, 2005) and also risks biasing them towards certain phenomena. Since it is not known what perceptual dimensions listeners use when rating the naturalness of synthetic speech, it is not clear what aspects of the signal it is possible to ask them to attend to. Therefore, a less direct methodology is adopted, where listeners are asked to perform a very simple task with the instructions containing no bias towards any particular acoustic property or perceptual dimension. This task is a simple "same or different" judgement on pairs of stimuli, from which we can derive a matrix of pairwise perceptual distances (based on the percentage of responses marked 'different'). Multidimensional scaling (MDS) allows such data to be visualised in a fixed number of dimensions. As discussed in Chapter 1, the MDS visualisation is a trade-off between using a large number of dimensions, which describe these distances between conditions in a high level of detail, and using a lower number of dimensions such that visualising these distances is practical. From this visualisation we can identify the perceptual dimensions which listeners are attending to. Tracing these back to the simulated effects involves interpreting the MDS visualisation.

3.3 Methodology

The aim is to tease apart the complex effects of statistical modelling on synthetic speech. In order for contributing factors to the shortcomings in quality of the speech output by HMM synthesis to be investigated, we need a framework in which these effects can be individually manipulated – a kind of 'oracle' HMM synthesiser which allows for complete control over each aspect of the system, varying it between some

form of 'ideal', or 'perfect' component and the real component used in a full HMM synthesiser. An obvious example of the 'ideal' is a vocoder, which has access to natural speech parameters and as such is unaffected by any flaws in the way the statistical modelling part reconstructs these. As such, this condition is used as the upper bound in the perceptual testing carried out in this chapter; however it should be noted that vocoding may introduce its own degradations on the quality of speech. Testing of the effect of vocoding is carried out in Chapters 4, 5 & 6.

3.3.1 Simulating "over-smoothing"

There are several ways in which the output speech parameters of an HMM synthesiser are "too smooth". Here, the concentration is on temporal effects, leaving spectral smoothness as work covered in Chapters 4 & 5. Looking at the output of typical HMM systems (Zen et al., 2009, Tokuda et al., 2013), there is generally far less temporal detail present than is observed in the speech parameters for natural speech. Some of this detail may simply be noise introduced by the spectral envelope estimation process, but some of it may be perceptually important. A simulation of the observed temporal smoothness from HMM synthesis is included in order to investigate the importance of temporal detail for speech parameters. This simulation is performed by temporally smoothing natural vocoded speech parameters. Temporal smoothing is present in HMM synthesis due to limited temporal resolution of 5-state-per-phone models and the subsequent MLPG trajectory generation algorithm (Tokuda et al., 2000, Yoshimura, 2002, King, 2011).

Another consequence of statistical modelling is that the variance of the generated speech parameters is lower than those from natural speech. This has long been known to significantly reduce the quality of the generated speech and is why mitigating this by considering Global Variance (GV) (Toda and Tokuda, 2007, Zen et al., 2009) has such a dramatic positive effect on quality. However, GV cannot guarantee to perfectly restore the correct variance of the parameters. The effect of modelling and of GV is simulated by scaling the standard deviation of the speech parameters by a value greater or less than 1.0.

Removing temporal detail via smoothing will also slightly reduce the variance of the speech parameters. The interaction between temporal smoothness and variance can be examined by applying both effects, with varying strengths. It is worth repeating at this point that temporal smoothing and variance scaling are certainly not a com-

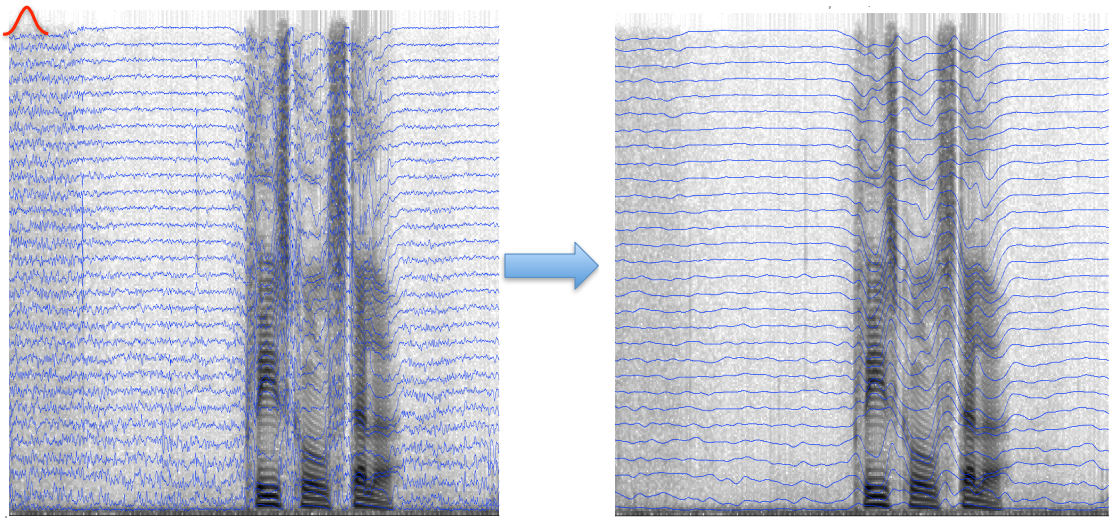


Figure 3.1: *Example of applying temporal smoothing to LSF parameteris using a sliding Hanning window.*

prehensive simulation of HMM synthesis, but they are used here as a starting point of the investigation and more complex effects will be investigated in Chapters 4, 5 & 6. The effects simulated in this investigation are all applied to each speech parameter coefficient independently and are implemented utterance-by-utterance.

3.3.1.1 Temporal smoothing

The smoothing effect was implemented as a weighted moving average, sliding a Hanning window over the signal (i.e., each LSF coefficient in turn), to simulate the limited temporal resolution of HMM modelling. The width of the window was varied, to impose varying amounts of smoothing. Figure 3.1 shows an example of this process. It should be noted that a side-effect of performing temporal smoothing is that the variance of the signal is also reduced. The impact of this side-effect will be investigated in Chapter 4.

3.3.1.2 Variance scaling

Variance adjustment was implemented as a simple scaling of the standard deviation by a fixed factor. For each parameter (i.e., each LSF) in turn, the mean value over the utterance was found and subtracted before multiplying the parameter by a scalar

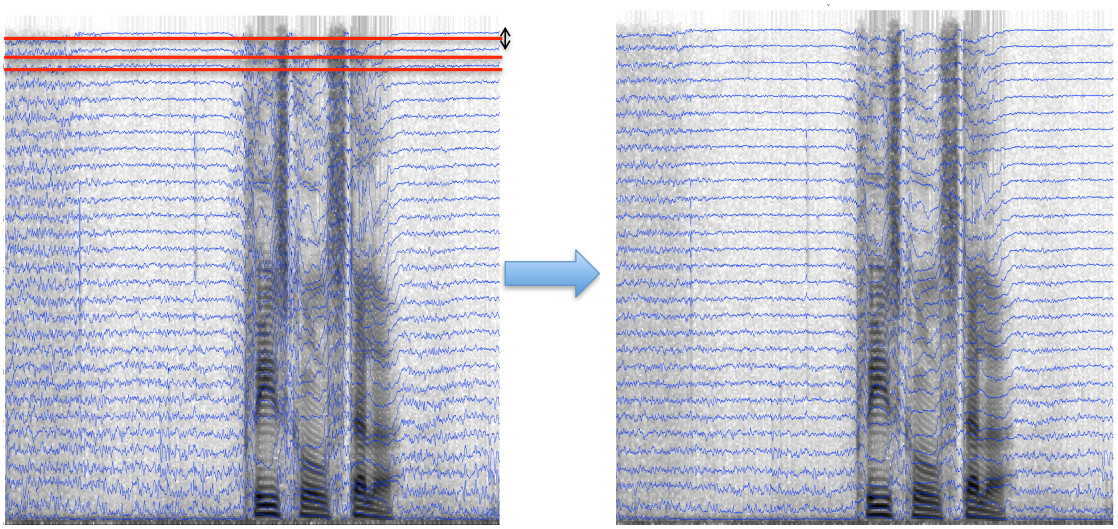


Figure 3.2: *Example of applying variance scaling to LSF parameters. In this example the variance is being scaled down.*

value, and finally adding the mean back in. By altering the scalar value, the standard deviation is correspondingly adjusted, to simulate both reduced variance (which is commonly observed in HMM synthesis) and increased variance (e.g., as may happen if a Gaussian p.d.f. is poorly estimated during training, or when GV fails to re-instate the appropriate amount of variance). This approach of variance scaling is similar to the postfiltering method investigated by Silén and Helander (2012). Figure 3.2 shows an example of this process.

3.3.2 Implementation

In this chapter, the concentration is on global simulations of the statistical modelling part of the system. This is illustrated in Figure 3.3, where we can see that the speech parameter extraction and waveform generation (reconstruction) parts are the same as in a full HMM synthesiser. Extraction of the spectral, f_0, and aperiodic energy speech parameters is performed as usual, with the use of the STRAIGHT vocoder (Matlab implementation)[1] (Kawahara et al., 1999, Liu and Kewley-Port, 2004). The extracted vocoder parameters were then processed using SPTK (Imai et al., 2012) to convert the spectral envelope to line spectral frequencies (LSFs), f_0 to log-f_0 and aperiodic en-

[1] STRAIGHT V40_007 methods were used. These were written by Hideki Kawahara.

ergy to band aperiodic energy. LSF's were chosen because they are more convenient for visualisation than, for example, Mel-generalised cepstra, and this should ease the interpretation of the results later. However verification of findings using LSF and Mel-generalised cepstra parametrisations of the spectral envelope is conducted in Chapter 4 in order to observe any specific effects due to the parametrisation used. The conversion of f_0 to log-f_0 and aperiodic to band aperiodic was also performed to simulate common modelling conditions of all speech parameters. These conversions result in better tracking of the effect that modelling has on the spectral envelope parameters by implementing a system which is more realistic. We also focus only on the spectral envelope speech parameters here. Simulations on aspects of statistical parametric speech synthesis systems linked to prosody, such as duration and f_0 modelling, throw open the investigation to a huge number of additional experimental variations being required. This is because research into prosody and the underlying semantics of speech is a huge field of research in itself. Therefore these parameters are predominantly tested under best-case conditions and the focus is largely on the quality of modelling of spectral features.

Following the application of the modelling simulations (having resolved any issues with impossible LSF parameter values resulting from the strengths of the simulations applied – to be discussed in Section 4.4.1), the LSFs, log-f_0 and band aperiodic energy parameters were converted back into spectral, f_0 and aperiodic energy speech parameters using SPTK (Imai et al., 2012) before performing the 'reconstruction' phase of HMM speech synthesis, by inputting the speech parameters into STRAIGHT (Matlab implementation) to obtain the synthesised speech waveform as output.

3.4 Experiments

A range of simulated effects were selected to be tested, with the strengths of modifications being selected by informal listening to reflect the sorts of imperfections encountered in many HMM synthesis systems. For the temporal smoothing, Hanning window sizes of 81 and 111 frames (at a frame rate of 5 msec) were selected, along with a 'no smoothing' condition. Smaller window widths (i.e., less smoothing) were found to produce negligible perceptual effects. Variance adjustment involved scaling the standard deviation by scalar values of 0.6, 0.8, 1.2 and 1.4 as well as a 'no variance adjustment' condition equivalent to scaling by 1.0. These particular values for smoothing and variance adjustment were selected to provide audibly different speech

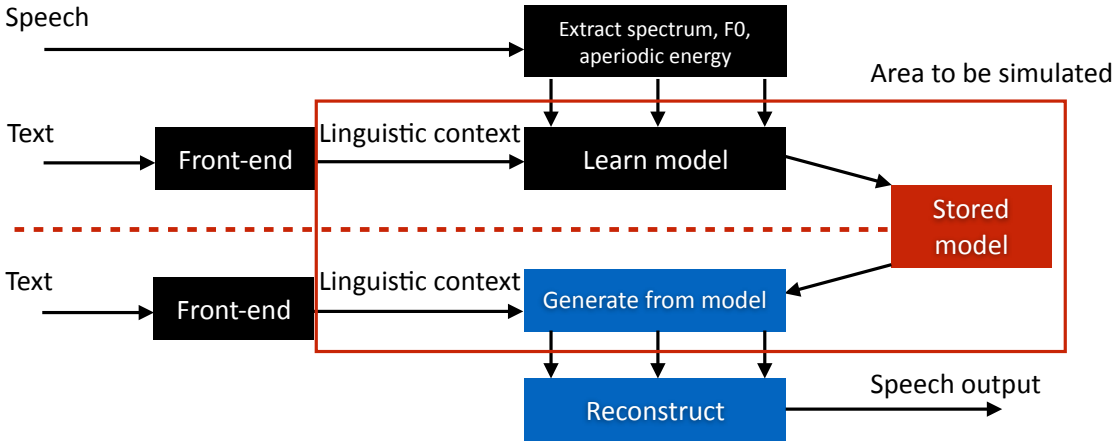


Figure 3.3: *Training and using an HMM speech synthesiser, illustrating the part of the process that is simulated here.*

quality, whilst staying within the range of qualities that are observed from real HMM synthesisers. As the investigation in this chapter has been performed to assess the inherent effects of temporal smoothing and incorrect global variance estimation, the strengths of the effect simulations selected are not necessarily indicative of the exact levels found in SPSS and should be considered more as a proof of concept. For example the temporal smoothing windows of 81 and 111 frames, correspond to a smoothing across time which is much larger than the state-level approximations commonly used in SPSS.

3.4.1 Materials

The speech corpus used for testing was a set of 40 Harvard Sentences (IEEE, 1969) read by a male professional speaker of British English (known as 'Nick' and whose speech has been used in the Hurricane Challenge (Cooke et al., 2013b,a) and who also features in the 'mngu0' acoustic-articulatory corpus[2] (Richmond et al., 2011)). The corpus was sampled at 16 KHz. It is worth noting that at the time work began on this thesis a sampling rate of 16 KHz for speech data was commonplace. Since then, higher sampling rates (48 KHz) have became commonplace, however it is believed that the findings from perceptual testing on 16 KHz speech still reflect the effects of the assumptions tested. Instead the overall quality of the speech is tracked to the sampling frequency used, with the quality across examples of fixed sampling frequency being

[2] http://www.mngu0.org

relative. The methodology for preparing the stimuli was, as described above, to extract speech parameters using STRAIGHT and SPTK, to apply the two simulated effects of smoothing and variance adjustment with all possible combinations of strengths including the 'no modification' conditions, then to reconstruct the waveform. Order 30 LSF coefficients were used as this offers a good representation of the spectral information for the speech at the sampling rate used. Given that there are 3 temporal smoothing conditions and 5 variance scaling conditions included in the listening test (as mentioned above), there are 15 possible combinations of these conditions.

The variance adjustment method was applied per speech parameter coefficient per utterance independently, so the mean speech parameter value subtracted before scaling is influenced by the amount of silence present; therefore, the material was manually edited to leave only just a few 100 msec of leading and trailing silence. Care was also taken to remove any background noise present during the non-speech[3], because in preliminary experiments this became perceptually much more apparent after applying some of the modifications. Although this investigation is done using a single speaker, a number of different speakers will be tested throughout Part 1 of the thesis, allowing for speaker-independent conclusions to be drawn.

3.4.2 Listening test

In the listening test, listeners had to make forced choice 'same or different quality' judgements about pairs of stimuli. The stimuli for testing was created by applying each of the 15 simulation conditions (called A to O) as defined in Table 3.1, which combine smoothing and/or variance adjustment to each of the 40 sentences. The 40 sentences were divided into 20 pairs (sentences 1 & 2, sentences 3 & 4, and so on), and for each of these pairs of sentences, all possible combinations of conditions (e.g., sentence 1 in condition A + sentence 2 in condition F) were created, except for pairs of identical conditions (e.g., sentence 1 in condition A + sentence 2 in condition A), as shown in Figure 3.4. These same condition comparisons were omitted as they provided little insight while adding a substantial number of additional responses required to be gathered from test subjects. Sentence-wise subjective testing is commonplace in speech synthesis, in order to assess the quality of synthesised speech (King and Karaiskos, 2012, 2013, Black and Muthukumar, 2015, Wester et al., 2015). As such sentence-wise subjective testing is conducted throughout the thesis.

[3]These edited waveforms were kindly provided by Catherine Mayo.

As stated above, all of the pairs of sentences presented to listeners are slightly different from each other, due to the removal of matching condition pair comparisons in order to reduce the required number of comparisons to be made. Therefore conducting pair-wise comparisons between conditions using the same sentence (as is more commonplace in speech synthesis perceptual testing) runs the risk that listeners will simply listen out for extremely fine local differences between conditions rather than rating the overall quality of the two sentences. This has two problems; firstly- if listeners respond different with every comparison this will not give us any overall information about which conditions are perceptually similar as this will place all conditions maximally far from each other, secondly- in this perceptual test we want listeners to inform us of the overall quality levels of the conditions rather than tiny local artefacts which may vary ever so slightly but not affect the overall quality. In order to stop these two points from occurring during perceptual testing, differing sentences were selected for the pair-wise comparison, requiring listeners to listen to the overall quality of the speech.

This resulted in $20 \times ((15 \times 15) - 15) = 4200$ pairs of sentences, which were then randomised in order and divided amongst 30 listeners, resulting in each listener listening to 140 pairs of sentences and thus making 140 'same or different' judgements. These listeners were selected at random from applicants to an online advert placed in the University of Edinburgh's Student And Graduate Employment service; all were native English speakers with no self-reported hearing problems. The stimuli pairs were presented in a randomised order per listener over high quality headphones in quiet sound-proofed booths with no distractions.

3.4.3 Multidimensional scaling

The raw listener responses were pooled across all listeners and all sentences for each individual combination of modifications. The result is a dissimilarity matrix, in which each cell contains a number indicating the perceived dissimilarity between two conditions. Figure 3.5 shows this matrix graphically: each cell contains the fraction of comparisons between a pair of conditions marked as 'different' by listeners. Multidimensional scaling was used to analyse this matrix, and create a plot in which each condition appears as a point. Short distances between points on the plot indicate perceptual similarity and large distances indicate dissimilarity (Borg and Groenen, 2005).

Table 3.1: *The 15 conditions combining each level of smoothing (including no smoothing) and each amount of standard deviation scaling (including no modification)*

Condition index	Hanning smoothing window size	Standard deviation scaling
A	none	0.6
B	81	0.6
C	111	0.6
D	none	0.8
E	81	0.8
F	111	0.8
G	none	none
H	81	none
I	111	none
J	none	1.2
K	81	1.2
L	111	1.2
M	none	1.4
N	81	1.4
O	111	1.4

		Sentence 1														
		A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
	A	×	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
	B	✓	×	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
	C	✓	✓	×	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
S	D	✓	✓	✓	×	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
e	E	✓	✓	✓	✓	×	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
n	F	✓	✓	✓	✓	✓	×	✓	✓	✓	✓	✓	✓	✓	✓	✓
t	G	✓	✓	✓	✓	✓	✓	×	✓	✓	✓	✓	✓	✓	✓	✓
e	H	✓	✓	✓	✓	✓	✓	✓	×	✓	✓	✓	✓	✓	✓	✓
n	I	✓	✓	✓	✓	✓	✓	✓	✓	×	✓	✓	✓	✓	✓	✓
c	J	✓	✓	✓	✓	✓	✓	✓	✓	✓	×	✓	✓	✓	✓	✓
e	K	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	×	✓	✓	✓	✓
	L	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	×	✓	✓	✓
2	M	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	×	✓	✓
	N	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	×	✓
	O	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	×

Figure 3.4: *One set of pairings of sentences and conditions in the listening test. Figure appeared in Merritt and King (2013).*

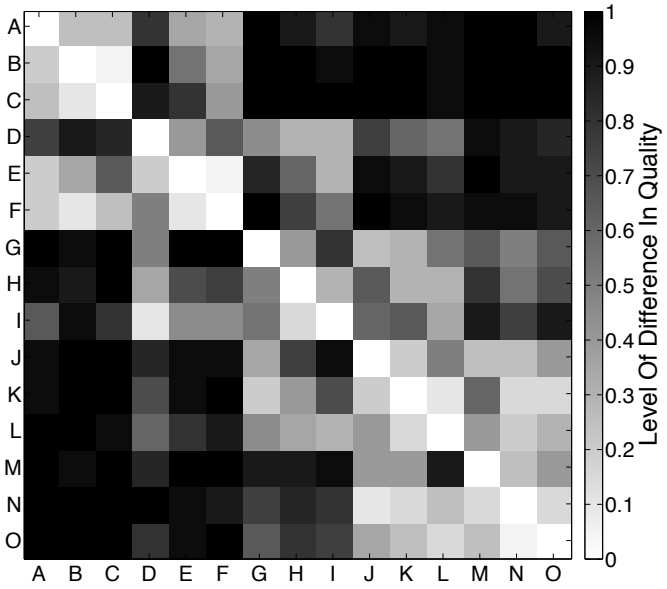


Figure 3.5: *Listeners' responses between conditions presented in Table 3.1, pooled across all sentences and listeners. Darker shades indicate greater perceived dissimilarity between conditions. Figure appeared in Merritt and King (2013).*

3.5 Results

MDS projects the dissimilarity matrix into a multi-dimensional space. As mentioned earlier, in order to find an appropriate dimensionality of this space, one must compromise accuracy of representation (in higher dimensions, the correspondence between dissimilarity and distance in the space will be more precise) against the need for a modest number of dimensions to allow for the data to be visualised and for the axes to be interpreted. The so-called stress value computed as part of the multidimensional scaling algorithm reflects this tradeoff; Figure 3.7 plots the stress value for various dimensionalities. Based on the principles for selecting an appropriate number of dimensions to visualise listener responses, described in Chapter 1, three dimensions were selected as a reasonable operating point.

The first two dimensions of the MDS are shown in Figure 3.6. Distance in this space indicates perceived dissimilarity: the closer a point is to the natural unmodified (vocoded) speech, the "more natural" it sounds. It is immediately apparent that the listeners judgements cannot be explained by a single dimension and that they are making their decisions based on more than one aspect of the speech:

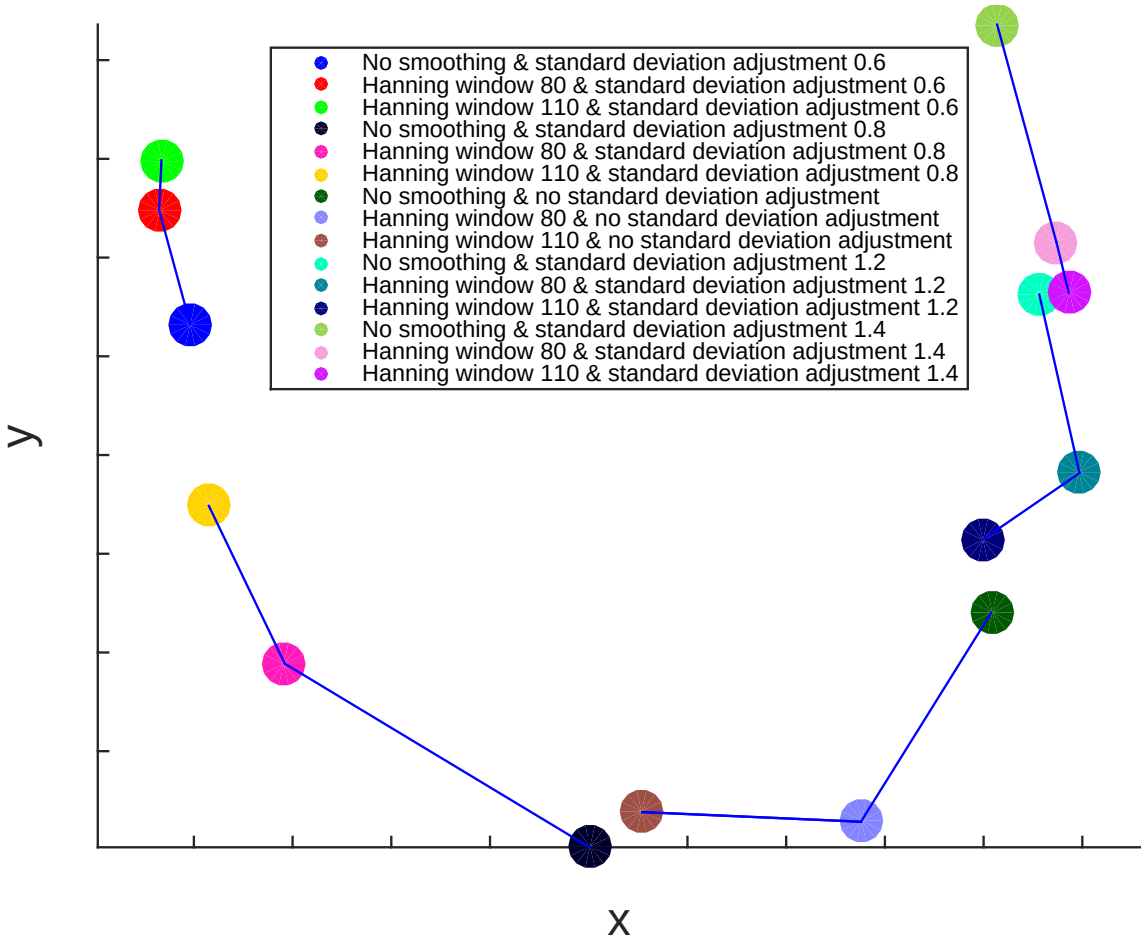


Figure 3.6: *Plot of the first two dimensions of the MDS. Lines have been added, connecting points with the same amount of variance modification but differing amounts of smoothing.*

- The horizontal axis seems to relate to the amount of LSF variance, with the reduced variance speech clearly different from the increased variance speech.

- The vertical axis seems to relate to overall quality of synthesis, regardless of the LSF variance, with both reduced and increase variance speech being placed towards the top of the space, whereas natural speech is at the bottom.

This plot also shows that the smoothing has only a secondary effect, probably simply because it has the side effect of slightly reducing variance. When the variance is too high (right hand side of Figure 3.6), then the smoothing has a beneficial effect, moving the points lower and therefore closer to natural speech.

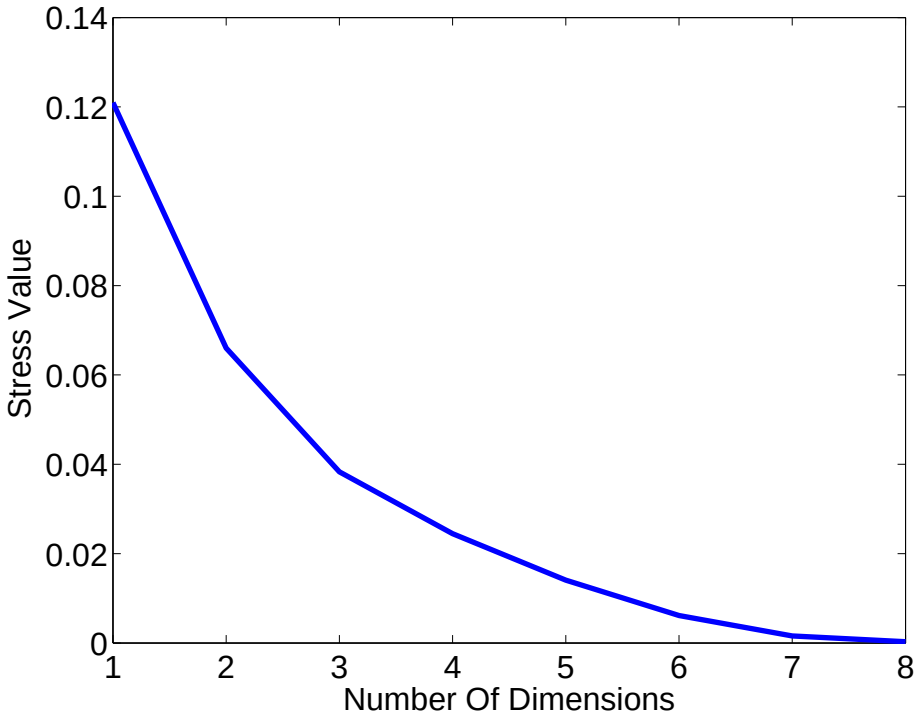


Figure 3.7: *Stress levels returned by MDS at different dimensions. Figure appeared in Merritt and King (2013).*

3.6 Conclusions

In this chapter, a simple-to-use, extensible methodology that can tease apart the contributions to synthesis quality of the various components of an HMM-based text-to-speech system has been introduced. The fundamental idea is to simulate all or part of the system, and thus to gain explicit control over the system's behaviour. This chapter has demonstrated the use of this framework in a straightforward way, by simulating a complete HMM-based synthesiser as simply a combination of smoothed parameter trajectories and incorrect variance.

Even from this very simple simulation, we can conclude that listeners are able to perceive different types of quality reduction: the MDS analysis reveals that they can make overall quality judgements (vertical axis of Figure 3.6) and at the same time clearly distinguish whether this is due to too high or too low variance. It also seems fairly safe to conclude that *temporal* smoothness in LSF trajectories is not really a problem and leads to only very small perceptual effects.

The methodology presented in this chapter will be further expanded in Chapters 4 & 5. These chapters will investigate additional hypothesised causes of reduced quality in statistical parametric speech synthesis along the continuum between natural speech and full synthesised speech.

Chapter 4

Attributing modelling errors in HMM synthesis by stepping gradually from natural to modelled speech

This chapter is an expanded version of the work in Merritt et al. (2015a) and therefore the text is closely related to that.

The work carried out in this chapter was part of a collaboration. The variance scaling and temporal smoothing simulation effects were implemented by myself, these were sent to Javier Latorre where they were recoded in order to be compatible with the system used at Toshiba Research and double checked by myself. Discussions of the systems to use in this investigation were carried out between myself, Simon King and Javier Latorre. The final selection of methods to be used for testing was made by myself. The code for running the listening tests described in Section 4.5 was my own as was the analysis of the responses. The code for running and analysing the MUSHRA listening test described in Section 4.4.1 was kindly provided by Gustav Eje Henter. The correction method for fixing problematic Mel-LSF coefficient values described in Section 4.4.1 was designed by myself and implemented into the Toshiba Research code by Javier Latorre.

4.1 Introduction

As discussed in Chapter 1, hidden Markov model (HMM) synthesis remains significantly behind the quality of natural speech and speech output from concatenative (unit selection) synthesis systems under 'best-case' conditions. Whilst the HMM approach

is relatively robust when it comes to handling training data with poor phonetic coverage or low recording quality (Zen and Toda, 2005), it fails to produce natural-sounding speech even when plentiful high-quality data is available (Tokuda et al., 2013).

Various explanations have been postulated regarding the cause of this apparent 'ceiling effect' in the level of quality achievable. The most common of these are: reduced variance of the spectral envelope as a consequence of averaging over multiple speech samples (Toda and Tokuda, 2007, King, 2011); over-smoothing of the parameter trajectories due to the MLPG algorithm (Tokuda et al., 2000); poor performance of vocoders (Zen et al., 2009), particularly regarding source-filter separation. However, before this thesis, these theories were rarely tested in formal studies.

The study in Chapter 3 introduced a methodology for testing hypothesised causes of degraded speech in HMM speech synthesis. This methodology involves viewing different hypothesised causes of reduced quality as individual elements within a continuum ranging from natural speech to standard HMM synthesised speech. The investigation in Chapter 3, however, was limited to adjusting temporal smoothness and variance of the speech parameters in vocoded speech. This chapter extends the methodology to add a number of novel contributions. The first of these contributions is the use of an idealised 'pseudo-HMM' condition. This condition involves averaging across the (few) contiguous frames from a single training example aligned with one HMM state. In doing so, the pseudo-HMM condition removes the effect of averaging across differing linguistic contexts and instead calculates an "ideal" model mean value across frames of matching linguistic contexts. The second of the contributions in this chapter is the repetition of perceptual testing on two different popular speech parametrisations under the range of conditions tested. This identifies potential parametrisation-specific findings from the investigation. The third of the contributions in this chapter is the use of a different vocoder; Toshiba's pitch-synchronous Fourier transform (PSFT) vocoder. Comparison with the findings of the STRAIGHT vocoder used in Chapter 3 allows for vocoder-specific findings to be identified. The fourth of the contributions in this chapter is the use of a commercial-quality speech database. The fifth of the contributions made in this chapter is the inclusion of natural (not vocoded) waveforms. The inclusion of natural speech allows for monitoring of the effect of vocoding on speech quality and allows perceptual findings within multidimensional scaling (MDS) space to be anchored. The sixth contribution of this chapter is the inclusion of a complete text-to-speech system. The inclusion of this full system allows for perceptual testing to place the conditions tested in relation to HMM synthesised speech.

As in Chapter 3, the focus is on spectral parameters, removing the much more convoluted question of prosody from the investigation. In all stimuli presented to listeners in this chapter, the natural phone durations (found using forced alignment) and f_0 were used.

4.2 Methodology

The speech continuum methodology introduced in Chapter 3 simulates various hypothesised causes of reduced quality as a result of modelling speech parameters in an HMM framework. Whilst the approach is general and extensible in principal, it was only used to investigate the perceptual effects of temporal smoothing and incorrect (too large or too small) variance in the trajectories of speech spectral envelope (i.e., filter) parameters. In this chapter the methodology is extended in terms of the modelling effects that are investigated. In brief, the methodology involves creation of various stimuli through simulations of HMM modelling effects. This is followed by a pairwise "same or different quality" listening test, and analysis of the responses is performed using MDS.

4.3 Creating the speech stimuli

All natural and vocoded speech samples were based on speech from a male speaker (*mgt*) from the Toshiba *Studio-HQ* database (Wan et al., 2014). This is a professional speaker recorded in a high quality studio, speaking in a neutral style. 1456 sentences from the same speaker were used to train the models. Details of the speech data from the *mgt* speaker, along with statistics of the models produced from this are detailed in Table 4.1[1].

From this table we can make interesting comparisons between the Mel-cepstra and Mel-LSF systems built. Firstly, performing decision tree regression to produce models using Mel-LSF parametrisation results in much larger decision trees being built than when decision trees are constructed using the Mel-cepstra parametrisation (i.e., the decision trees have a larger number of leaves). As a result of this, the number of linguistic contexts from the training data present in each of the leaves in the decision tree under the Mel-LSF parametrisation is much fewer than is the case under Mel-cepstra parametrisation. The reason exactly why this is the case is not obvious.

[1] Statistics were collected by Javier Latorre.

Further investigation into this is left as future work. Additionally, it is interesting that the number of linguistic contexts present in the models selected for use in the test utterances is much higher than the median number of linguistic contexts in the leaves of the overall decision tree. This indicates that there is a large imbalance in the number of linguistic contexts present in different leaves in the decision tree, presumably resulting in some leaves being sparsely populated with more rare linguistic contexts as a result of the recording script used to record the corpus (i.e., as a result of using phonetically balanced utterances).

4.3.1 Speech parameters

Spectral parameters were extracted with a Fourier transform using pitch synchronous windowing. The parameters are however estimated at fixed-frame. This output was then transformed into either Mel-Cepstral or Mel-line spectral frequency (LSF) coefficients (Tokuda et al., 1994) using SPTK (Imai et al., 2012). The aperiodic energy was estimated using a pitch-scaled harmonic filter (Jackson and Shadle, 2001) and parametrised into 23 bark-scaled aperiodicity bands.

The experiments in this chapter are run once using Mel-Cepstral coefficients and again separately using LSF coefficients as these are two popular parametrisations of filter coefficients within statistical parametric speech synthesis. By re-running the experiments on each of these parametrisations, any parameter-specific effects under each of the conditions tested can be observed.

4.3.2 Simulating the effects of modelling

The standard approach to statistical parametric speech synthesis uses HMMs with a fixed number of emitting states (Zen and Toda, 2005), each containing a multivariate Gaussian distribution. When generating from such a model using the MLPG algorithm (Tokuda et al., 2000), a sequence of frames is emitted from each state: the mean of those frames is constant over the duration of the state. This introduces an effect of *temporal smoothing* over the generated parameters, and the amount of smoothing varies with the state duration.

The mean values associated with each state are estimated from data by averaging (typically via Expectation-Maximisation) the speech parameters from the contiguous sequence of frames associated with that state. This introduces *averaging across examples of matching linguistic context*. Furthermore, since no training database can

Table 4.1: *Statistics of the models produced on speaker mgt from Toshiba Studio-HQ database used for testing.*

# contexts in training	All	80958	
	excluding silence	78905	
# contexts in evaluation	All	1684	
	excluding silence	1650	
Frames (including silence)		1,590,000	
Parametrisation		Mel-Cep	Mel-LSF
# leaves	state 1	302	593
	state 2	392	726
	state 3	354	726
	state 4	302	573
	state 5	331	543
	total	1681	3261
Median # context/leaf in training	state 1	210	96
	state 2	127	70
	state 3	141	69
	state 4	179	89
	state 5	169	86
Median deviation in training	state 1	112	56
	state 2	78	46
	state 3	89	45
	state 4	113	59
	state 5	110	52
Mean # training context/leaf for evaluation context	state 1	283.4	162.4
	state 2	239.0	143.3
	state 3	257.0	145.4
	state 4	296.9	175.3
	state 5	274.7	152.4
Median # training context/leaf for evaluation context	state 1	222.5	120
	state 2	160	101
	state 3	176.0	106
	state 4	199	123
	state 5	186.5	109

include sufficient examples of every class (i.e., every unique linguistic context string), examples drawn from differing contexts must be pooled and averaged together in order to robustly estimate the state mean and variance. This introduces *averaging across differing linguistic contexts.*

In addition to this, the *variance* of the trajectories generated from model outputs in HMM synthesis may not match that of natural speech. This could be due to the estimation of the model parameters from limited data, and/or inadequate models, and/or the parameter generation method.

4.3.2.1 Temporal smoothing

The temporal smoothing effect was implemented in the same way as in Chapter 3, to simulate the temporal smoothness of speech parameters generated by MLPG. This effect is simulated by calculating a weighted moving average across the signal (implemented by sliding a Hanning window over the signal, each coefficient in turn).

4.3.2.2 Variance adjustment

The variance adjustment effect was implemented in the same way as in Chapter 3, to simulate the potentially-incorrect variance of generated trajectories (which can occur even when the GV technique (Toda and Tokuda, 2007) is employed). This is implemented by subtracting the utterance-level mean for each coefficient from its respective coefficient trajectory. The resulting trajectory is then multiplied by a scalar value to increase or decrease the signal variance. The mean value is then added back to the signal. Previous investigations (Silén and Helander, 2012) have found that this variance-scaling method can enhance the speech as much as GV.

A new condition for observing the effect of variance adjustment has been added to the investigation in this chapter. This is to restore the level of variance across the utterance per-coefficient to the original variance level following the use of the temporal smoothing effect (this may result in a different scalar factor being used to adjust the variance for each coefficient). The addition of this condition aims to piece apart the effect of temporal smoothing from the effect of a loss of signal variance, which is a side-effect of the temporal smoothing step.

4.3.2.3 Parameter averaging

It was hypothesised that the effect of averaging over short sequences of contiguous frames from a single training example (i.e., frames belonging to the same linguistic context) is small compared to averaging across frames drawn from differing contexts (as occurs following decision tree regression in HMM synthesis). To test this, we constructed an idealised "pseudo-HMM" in which only averaging across frames within the same linguistic context is present. For comparison, we also used a complete, speaker-dependent HMM system similar to that described in Zen and Gales (2011), which of course does involve both averaging across frames within matching linguistic contexts and averaging across frames drawn from different contexts.

The pseudo-HMM is created by using a natural example of the sentence to be 'synthesised', to ensure that the contexts are an exact match. For each such individual utterance, an association between states and frames was obtained by forced alignment using a speaker-dependent HMM. The mean value of each state was computed as the median[2] of the frames associated with that state. The variance values from the standard HMM system were used alongside this 'ideal' model mean value.

During the synthesis with either HMM or pseudo-HMM, the phone and state durations of the corresponding natural utterance were used. For the excitation signal, the original f_0 was first made continuous by interpolating through unvoiced regions, then combined with the aperiodicity values to generate mixed-excitation (Latorre et al., 2011). In the case of the HMM, the aperiodicity values were those generated by the model, to ensure consistency with the generated spectral envelope. The consistency between aperiodicity and spectrum is important as they are strongly related. Even with continuous f_0, if a voiced spectrum is mixed with an unvoiced aperiodicity the result is a harsh noise. To avoid such mismatch affecting the judgements, synthetic aperiodicity was used with synthetic spectrum.

[2]Median was used instead of mean because it is more robust when the number of frames is small, which is the case here.

4.4 Implementation

4.4.1 Solving stabilisation issues with LSF coefficients

Following the implementation of the conditions to be included in this investigation it is possible for the LSF coefficients to acquire problematic values. For example the conditions to be tested can lead to coefficient trajectories which are too close together, cross each other, or exceed the Nyquist frequency. SPTK's 'lspcheck' function attempts to fix these issues by swapping values at crossing boundaries (Imai et al., 2012). However following the use of this function there were still a significant number of artefacts present. These caused there to be no speech produced at these crossing points or the lspcheck function to fail as it was unable to cope with these values. In order to remove these artefacts, Mel-LSF values were limited to be in the range between 0 and π rad, whilst maintaining a reasonable spacing. Between coefficients and within the limits of the interval an arbitrary small 'spacing' value of 0.01 rad was used. Wherever two coefficients crossed (i.e., were not in ascending order), the lower coefficient was reduced so that it was at least 0.01 rad lower than the higher-order coefficient. Following this, SPTK's lspcheck function was applied to fix any remaining issues.

4.4.1.1 MUSHRA testing

In order to test that these corrections successfully remove all audible artefacts while not introducing further artefacts, formal listening tests were performed. These tests were performed within the MUSHRA paradigm, where all conditions under a single sentence are presented side-by-side thus allowing the listener to make use of the full range of conditions when performing their judgements. The conditions included in this listening test are shown in Table 4.2. These conditions were selected to represent a reasonable range of the conditions which are required for the final listening test to be conducted in the investigation in this chapter. These conditions range from natural vocoded speech (V*) through to full HMM synthesis (H*), in order to verify that the proposed corrections don't have adverse effects. As described in Chapter 1, a natural recording (condition *N*) of each of the utterances tested is provided as a hidden reference for the listener and is present among the 11 conditions to be rated. Listeners are informed that this natural recording should be rated as 100. 20 native English speaking participants with no known hearing impairments were used for testing. Each participant was asked to rate the same 30 screens of sentences read by the *mgt* speaker. Each of these screens features the full range of conditions included in the listening test for a single sentence.

Table 4.2: *The 11 conditions included in the MUSHRA test*

Index	Condition index in Table 4.3	Correction method
N	Original	N/A
VC	Vocoded	SPTK lspcheck
VP	Vocoded	proposed correction & SPTK lspcheck
PC	pseudo-HMM	SPTK lspcheck
PP	pseudo-HMM	proposed correction & SPTK lspcheck
HC	HMM-synth	SPTK lspcheck
HP	HMM-synth	proposed correction & SPTK lspcheck
SC	hann-1-stddev-120	SPTK lspcheck
SP	hann-1-stddev-120	proposed correction & SPTK lspcheck
MC	hann-21-stddev-match	SPTK lspcheck
MP	hann-21-stddev-match	proposed correction & SPTK lspcheck

4.4.1.2 Findings from Mel-LSF correction method regression testing

Figure 4.1 shows the absolute values given by listeners to each of the 11 conditions tested. Figure 4.2 shows these scores in direct comparison between the proposed solution for fixing discrepancies in Mel-LSF values and the use of the SPTK lspcheck function alone. All tests for significant differences used Holm-Bonferroni correction due to the large number of condition pairs to compare. All like-for-like condition pairs under the two LSF-correction methods being tested (same condition but applying a different LSF-correction method) are significantly different from each other in terms of absolute value, except between VC and VP. Significant differences are in agreement using a t-test and Wilcoxon signed-rank test at a p value of 0.01. The t-test is a standard method of testing for significant differences between two conditions; the paired t-test is used in the thesis for identifying significant differences. The t-test assumes the data is normally distributed. Since it is the paired t-test being used, this data is the difference between pairwise listener responses. The paired t-test measures the likelihood that the differences between the two conditions fits the t-distribution (normally distributed) and therefore the likelihood that there is no significant difference between them. The Wilcoxon signed-rank test uses information about the magnitude between pairs of data as well as the sign (which item in the pair is scored higher). This significance testing does not make the assumption that the data fits a normal distribution (as is the case with the t-test) and instead records the pairwise distances between the

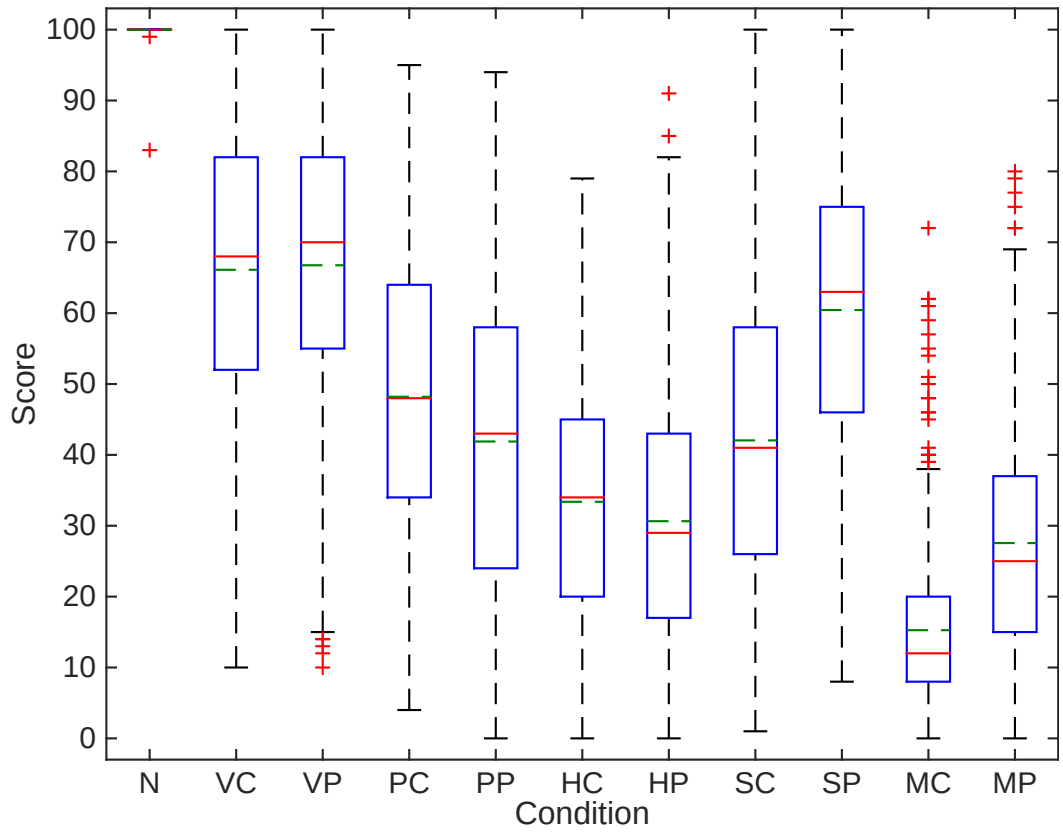


Figure 4.1: *Boxplot of absolute values given in MUSHRA test for LSF correction. The notation of the plot is as follows: the horizontal red lines show the median response values, the horizontal dashed green lines show the mean response values, the blue boxes show the 25th and 75th percentiles of the data, the whiskers show the range of responses excluding outliers, red crosses show outlier responses.*

responses from the conditions being compared. As the Wilcoxon signed-rank test is a pair-wise test of rank difference, this makes this test useful for testing for significant differences in not only absolute values, as tested here, but also for rank-order results, which will be used later in the thesis. As the t-test and Wilcoxon signed-rank test are both pairwise tests, they are particularly well-suited to processing responses from MUSHRA testing, given that the two samples which form the data pair, were in fact judged at the same time on the same utterance. The agreement between these tests are illustrated in Figure 4.3. Given that the only difference between the conditions shown in Figure 4.2 is the method of fixing discrepancies in the Mel-LSF values it seems that the proposed method fixes dramatic artefacts in the speech produced under the simulation conditions. Following this finding, the proposed method will be used to fix all discrepancies found in the Mel-LSF values.

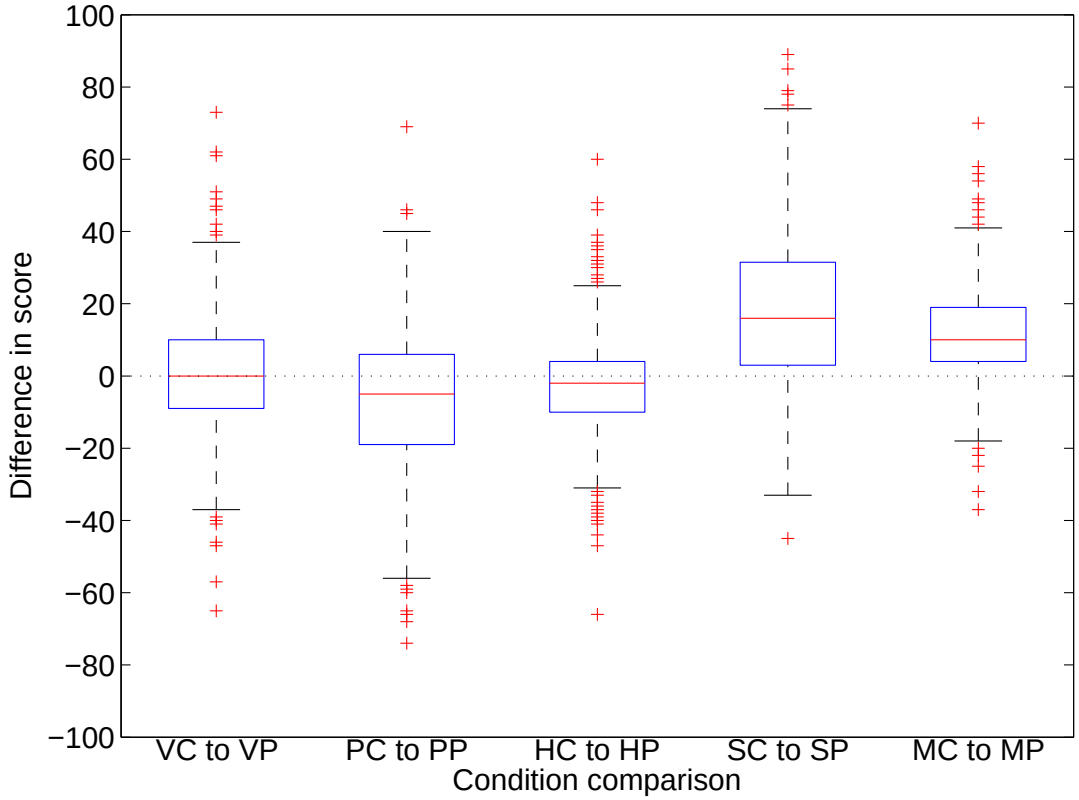


Figure 4.2: *Boxplot of the difference in absolute values given in MUSHRA test for LSF correction between conditions. The notation of the plot is as follows: the horizontal red lines show the median response values, the blue boxes show the 25th and 75th percentiles of the data, the whiskers show the range of responses excluding outliers, red crosses show outlier responses.*

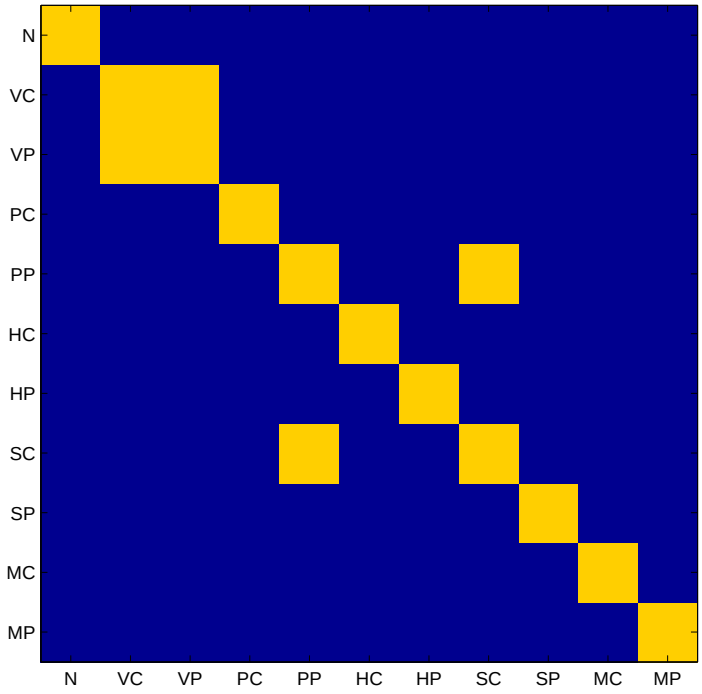


Figure 4.3: *Visualisation of significant differences between systems in terms of absolute value using t-test and the Wilcoxon signed-rank test (p=0.01). Dark blue indicates agreement in significant difference. Yellow indicates agreement in no significant difference.*

4.5 Experiments

The strengths of the temporal smoothing and variance adjustment modifications to be included in the listening test were chosen by informal listening, so that they created a similar range of imperfections to those found in the speech from the *HMM-synth* condition. The values chosen to scale the utterance-level variance were 80%, 100% (i.e., no modification), 120% and 140%. In addition, a variance adjustment condition using a scaling value chosen such that the final utterance-level standard deviation matched that measured before the application of temporal smoothing (*stddev-match*) was also included. The Hanning window sizes selected to represent various levels of temporal smoothing were (in terms of the width in frames); no smoothing, 5, 11 and 21. The full range of conditions presented to listeners for pairwise comparison is shown in Table 4.3.

In the listening test, listeners were asked to make forced-choice 'same or different quality' judgements about pairs of stimuli. 30 sentences taken from the same speaker used to train the models (see Section 4.3) were used for testing[3], to which each of the 22 selected conditions in Table 4.3 were applied. The two items in each comparison pair were differing sentences (randomly selected from the 30) under differing conditions (all possible pairs of conditions were covered, with the exception of comparing matching conditions). Every pair of stimuli was presented a total of 15 times. This resulted in a grand total of 6930 pairwise comparisons. Each of the 45 listeners in the test was asked to make 154 'same or different quality' judgements, selected randomly without replacement from the 6930 pairs. This number of judgements is within the limit which an individual listener can tolerate (Mayo et al., 2011).

The entire listening test was run twice: once using stimuli constructed using Mel-cepstra parameters, then again using Mel-LSF. In both listening tests, the *Original* speech waveform was also included. The outcome of each test is a matrix in which each cell contains the number of 'different' judgements, summed across listeners: i.e., a matrix of perceptual distances. Multidimensional scaling (MDS) (Borg and Groenen, 2005) is then used to visualise this matrix of distances, where each condition is a point in multi-dimensional space and distances between points reflect the perceptual distances from the data.

[3]In Merritt et al. (2015a) these were incorrectly reported to be 30 held-out sentences, however 15 were present in the training set while the remaining 15 were held-out. This is not expected to have much of an impact on the findings.

Table 4.3: *The 22 conditions presented to listeners*

Condition	Speech signal origin	Hanning smoothing window duration (frames)	Standard deviation scaling (%)
hann-1-stddev-080	vocoded	none	80
hann-5-stddev-080	vocoded	5	80
hann-11-stddev-080	vocoded	11	80
hann-21-stddev-080	vocoded	21	80
Vocoded	vocoded	none	100
hann-5-stddev-100	vocoded	5	100
hann-11-stddev-100	vocoded	11	100
hann-21-stddev-100	vocoded	21	100
hann-1-stddev-120	vocoded	none	120
hann-5-stddev-120	vocoded	5	120
hann-11-stddev-120	vocoded	11	120
hann-21-stddev-120	vocoded	21	120
hann-1-stddev-140	vocoded	none	140
hann-5-stddev-140	vocoded	5	140
hann-11-stddev-140	vocoded	11	140
hann-21-stddev-140	vocoded	21	140
hann-5-stddev-match	vocoded	5	match original
hann-11-stddev-match	vocoded	11	match original
hann-21-stddev-match	vocoded	21	match original
HMM-synth	HMM (with GV)	none	100
Original	natural	N/A	N/A
pseudo-HMM	pseudo-HMM	none	100

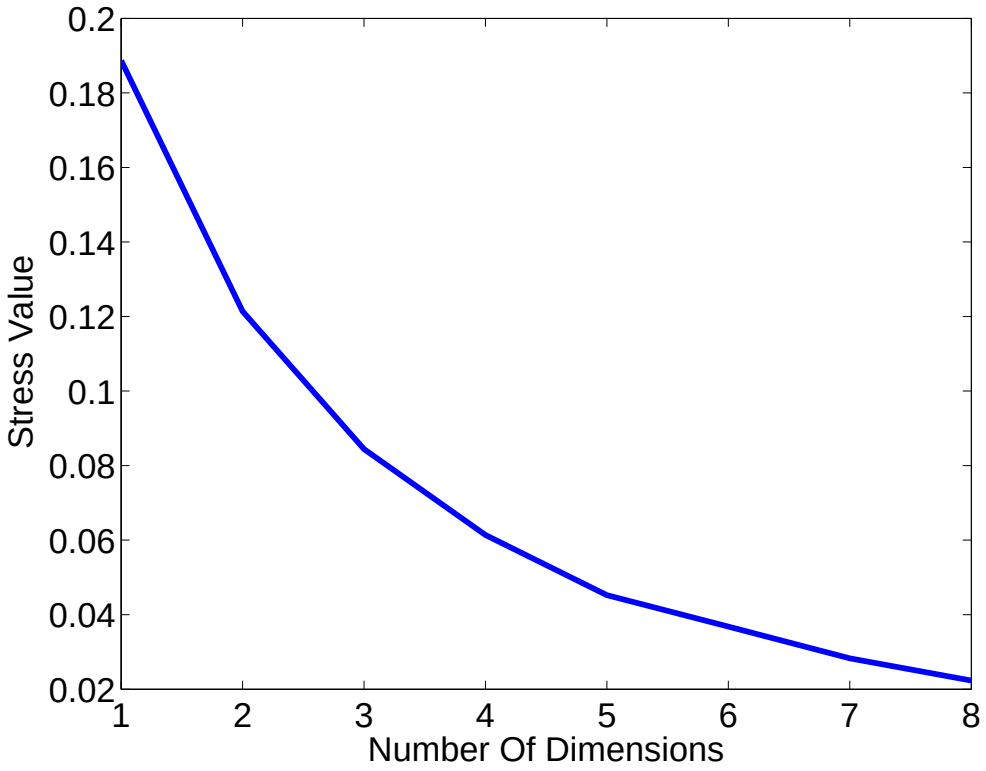


Figure 4.4: *Stress levels returned by MDS when attempting to fit the responses in the mcep listening test to different numbers of dimensions.*

4.6 Results

4.6.1 Mel-cepstral parametrisation

The stress levels for MDS at differing numbers of dimensions are shown in Figure 4.4. Based on the principles for selecting an appropriate number of dimensions to visualise listener responses, described in Chapter 1, three dimensions were selected as a reasonable operating point. Here we examine this space, two dimensions at a time.

Figure 4.5 plots the 2-dimensional X-Y projection of the 3-dimensional MDS space. Scaling the standard deviation of the speech parameters appears to correspond to a lower-right to upper-left movement in this MDS space. As variance is scaled from 80% to 140%, the speech becomes first closer to natural speech (at 100% and 120%) and then eventually moves further away, as we would expect. The *hann-5-stddev-match* condition, which is the same as *vocoding* (which we could also denote *hann-1-stddev-match*) but with very light smoothing applied, comes approximately as close to natural speech as vocoded speech does. This suggests that removing the fine temporal detail in speech parameter trajectories is not detrimental: it is probably noise arising

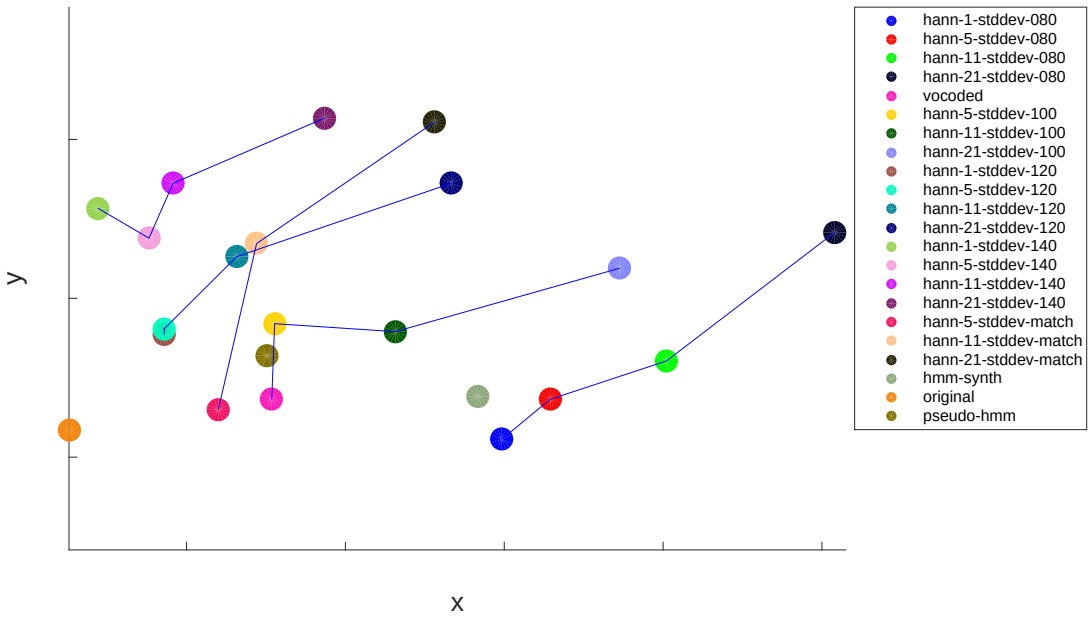
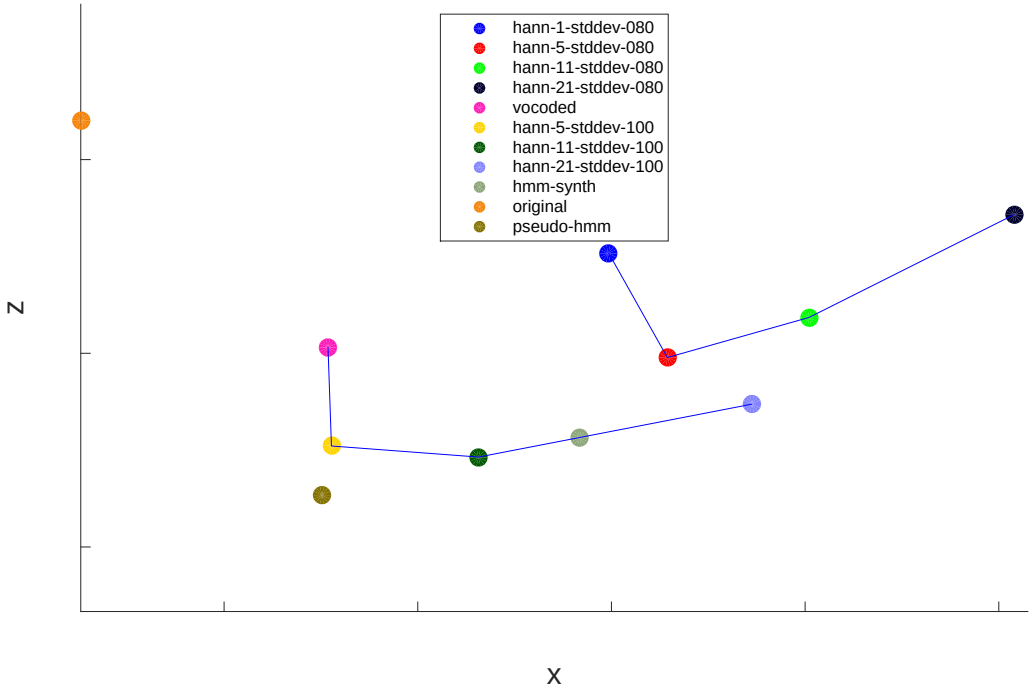


Figure 4.5: *X-Y projection of the Mel-Cepstral MDS space. Lines have been added, connecting points with the same amount of variance modification but differing amounts of smoothing. The point for natural speech is in the lower left corner. We can infer that points closer to this correspond to more natural-sounding speech. Figure appeared in Merritt et al. (2015a).*

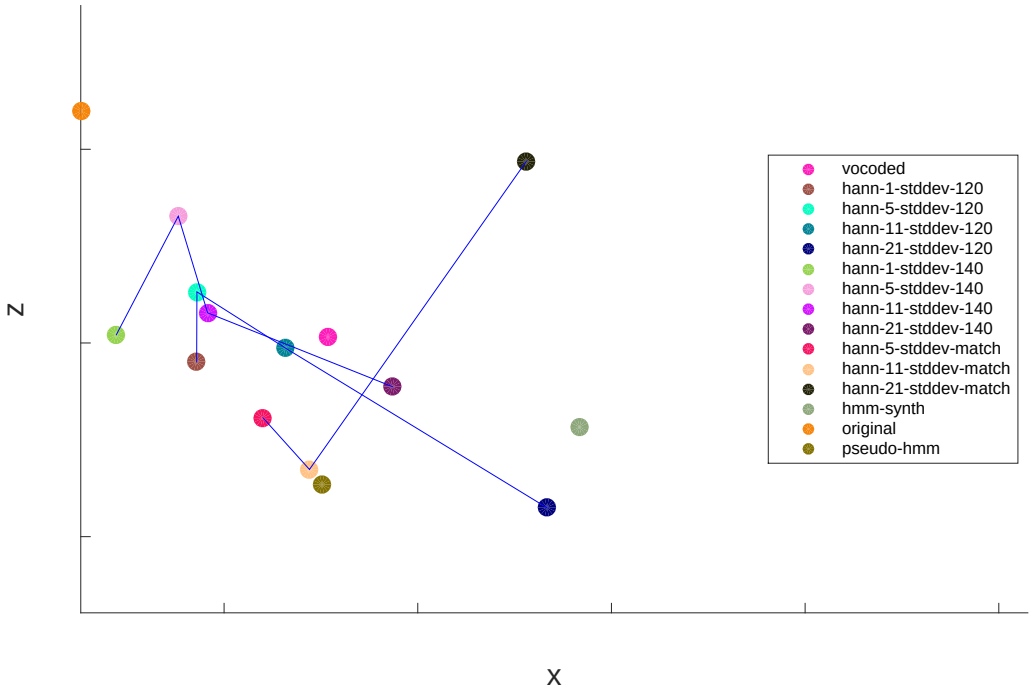
from parameter estimation, not speech information. Excessive smoothing (*hann-21-...*) pushes the speech quality far away from natural.

Figure 4.6 shows the X-Z projection of the MDS space, with the reduced and increased variance conditions plotted separately, for clarity. In Figure 4.6a, the points move towards HMM speech as a small amount of smoothing is applied and then move away again. Figure 4.6b reveals that applying increasing amounts of smoothing to the increased variance conditions initially gets us closer to natural speech before then moving away.

The pseudo-HMM condition is fairly close to conditions with variance unscaled (100%) or variance restored to the vocoded speech following light or moderate temporal smoothing. It is also considerably closer to vocoded speech (which is an upper bound for the pseudo-HMM) than the true HMM condition (*hmm-synth*). Together, these suggest that averaging across differing linguistic contexts is indeed more harmful to naturalness than averaging across frames with matching linguistic context. A side-effect of the pseudo-HMM condition introduced here is that the frames which are averaged together to provide this ideal model only ever come from a single example



(a) *Reduced variance data points, plus references.*



(b) *Increased variance data points, plus references.*

Figure 4.6: *X-Z projection of the Mel-cepstral MDS space. Figure appeared in Merritt et al. (2015a).*

of the matching linguistic context. Further investigation into whether optimal performance is observed as a result of averaging together frames whose linguistic context

exactly match or whether a further increase in quality is observed as result of these frames only coming from a single occurrence of a matching linguistic context (as occurs in the pseudo-HMM condition in this chapter) is of interest. However, given the low likelihood of an exact linguistic context match occurring in a training corpus of speech, it is not an unlikely scenario that such a model would predominantly be trained on only one example of a linguistic context.

4.6.1.1 Mel-cepstral conclusions

Matching the variance of natural speech is important. Whilst too much or too little variance both sound less natural, it would appear to be better to have slightly too much variance than too little. Light smoothing appears not to be detrimental, presumably because it merely removes minor artefacts created during parameter extraction (vocoding) from the original speech signal. Apart from getting the variance in the right range (equal to or slightly greater than that of vocoded speech), averaging across differing linguistic contexts is the single biggest cause of reduced naturalness of HMM synthetic speech.

4.6.2 Mel-LSF parameterisation

The stress levels for MDS at differing numbers of dimensions are shown in Figure 4.7. Based on the principles for selecting an appropriate number of dimensions to visualise listener responses, described in Chapter 1, two dimensions were selected as a reasonable operating point. The MDS projection using two dimensions is shown in Figure 4.8.

Figure 4.8 shows that conditions with slightly increased variance (standard deviation scaled to 120%), conditions with variance matching that of vocoded speech, and vocoded speech itself, are all perceptually about the same distance from natural speech. However, excessive variance (standard deviation scaled to 140%) is highly detrimental. As with the Mel-cepstral case, light smoothing does no harm, but heavy smoothing causes large reductions in naturalness.

Reducing the variance of the Mel-LSF parameters (standard deviation scaled to 80%) quickly moves the speech a large perceptual distance away from natural and vocoded speech. The HMM speech (*hmm-synth*) lies very close to some of the conditions with reduced variance (standard deviation scaled to 80%) and light to moderate smoothing, suggesting that the HMMs (despite the use of GV) fail to generate speech

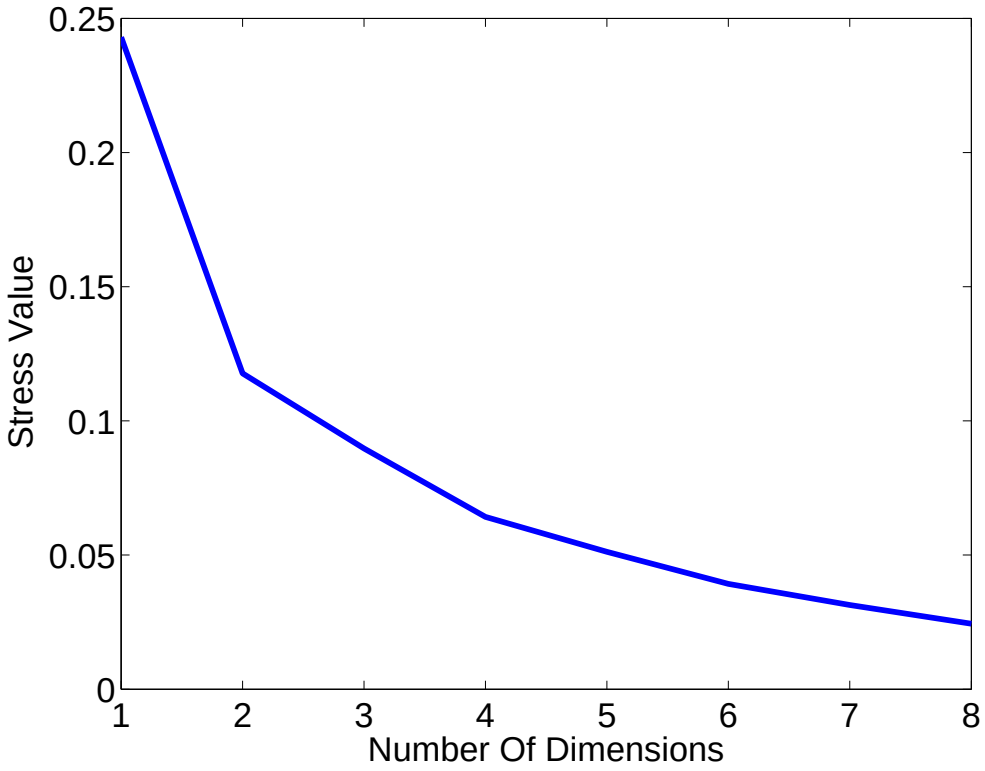


Figure 4.7: *Stress levels returned by MDS when attempting to fit the responses in the Mel-LSF listening test to different numbers of dimensions.*

parameters with adequate variance. The pseudo-HMM condition is considerably closer to vocoded speech than the true HMM condition (*hmm-synth*), which suggests again that averaging across differing linguistic contexts is indeed harmful to naturalness.

4.6.2.1 Mel-LSF conclusions

In the case of Mel-LSF parameterisation, too little variance is highly damaging and it is clearly better to err on the side of slightly more variance (than vocoded speech), than too little variance. Light smoothing is found to not be problematic. As with the Mel-cepstral parameterisation, averaging the contiguous sequence of frames aligned with a single HMM state (*pseudo-hmm*) degrades the speech a little, but not as much as averaging across different contexts (*hmm-synth*).

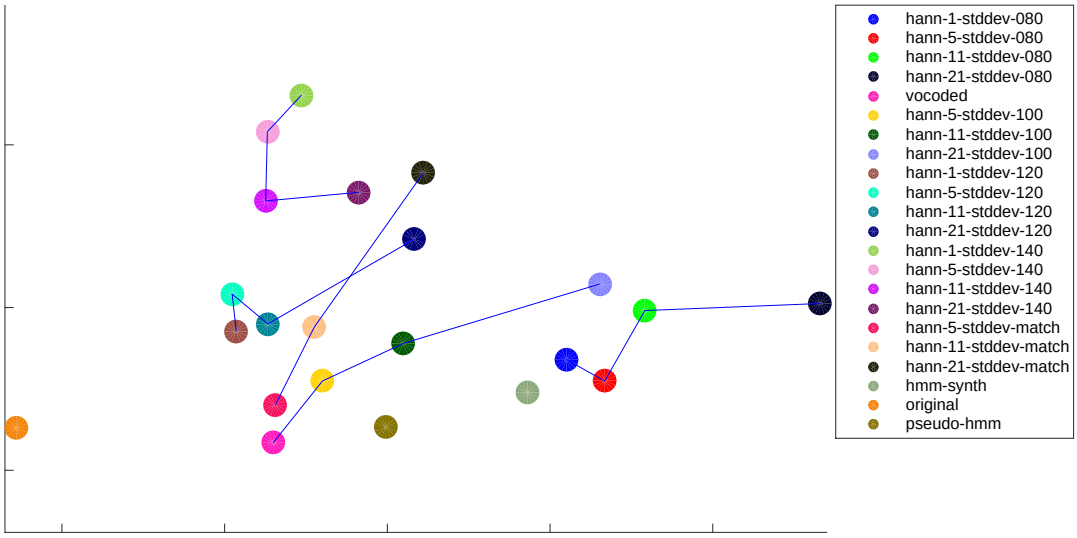


Figure 4.8: *The Mel-LSF MDS space. Lines have been added to aid readability, as in Figure 4.5. Figure appeared in Merritt et al. (2015a).*

4.7 Summary

The simulation framework introduced in Chapter 3 has been extended to compare a much wider range of conditions. The experimental results lead us to draw the following conclusions.

Generating speech parameters with the correct variance is preferred, as was found to be the case in Chapter 3. However this investigation is also able to extend these findings with the knowledge that erring on the side of slightly too much variance is much better than too little.

Small amounts of temporal smoothing are not harmful, as was found to be the case in Chapter 3. This finding is further strengthened by the perceptual responses for conditions where the utterance-level variance lost as a result of temporal smoothing is reinstated.

Averaging across examples with matching linguistic contexts (in this investigation this was calculated across short contiguous sequences of frames) was found to be mildly harmful, in a perceptually similar way to light temporal smoothing. Conversely averaging across differing linguistic contexts (as is performed in models in standard HMM synthesis systems following decision tree regression) was found to be very harmful.

The parametrisation which is used for the voice filter (Mel-cepstra or Mel-LSF) was found to not have much baring on the effect of modelling. Similar effects were

observed across both of the popular parametrisations tested.

The methodology for investigating hypothesised causes of reduced quality in HMM speech synthesis, introduced in Chapter 3, will be further extended in Chapter 5 to investigate the effects of modelling on source and filter components of speech. Chapter 5 will also investigate the effectiveness of current post-generation enhancement methods for SPSS, at 'un-doing' the effects of modelling.

Chapter 5

Investigating source and filter contributions, and their interaction, to statistical parametric speech synthesis

This chapter is an expanded version of the work in Merritt et al. (2014) and therefore the text is closely related to that.

The work in this chapter was done as part of a collaboration with Tuomo Raitio with the following division of work. The variance scaling and temporal smoothing simulation effects were implemented by myself, these were sent to Tuomo Raitio who recoded them in order to be compatible with the GlottHMM synthesis system. Decisions on the systems to be included in this investigation were made by myself and Tuomo Raitio. The code to synthesise the speech in GlottHMM was run by Tuomo Raitio. The code for running the listening tests was my own, as was the analysis of the responses.

5.1 Introduction

Chapter 3 introduced a methodology for piecing apart various hypothesised causes of the ceiling effect in quality observed within HMM speech synthesis. This methodology separates out different hypothesised causes of reduced quality by viewing these as separate elements within a 'continuum of speech', ranging from natural speech at one end through to full HMM synthesised speech at the other. This methodology al-

lows for individual formal testing of hypothesised causes of reduced synthesis quality from SPSS systems. Such an approach is more effective than speculating as to the contributing factors of different hypothesised causes following the use of a full HMM synthesis system, where their effects are co-occurring. Chapter 3 used this methodology to test the effect of temporally over-smooth trajectories, as output following the use of MLPG, and the effect of incorrect variance estimation of generated speech parameters. Chapter 4 extended the methodology to explore the effect on synthesis quality of averaging across differing linguistic contexts, a common occurrence within decision tree regression-based HMM synthesis. Chapter 4 also looked at the effect of the various elements tested within the 'continuum of speech' when using different parametrisations of speech (Mel-cepstra and Mel-LSF).

The methodology used in Chapters 3 & 4 will be further extended in this chapter to add a number of novel contributions. The first of these is to investigate the effect of independent modelling of speech parameter streams on the naturalness of the subsequent speech produced. This is a common occurrence in HMM synthesis, where typically each speech parameter stream is modelled by a separate decision tree. By using the GlottHMM vocoder we are able to perceptually test the effect of modelling on source parameters and filter parameters individually. The second contribution of this chapter is the use of a different vocoder; GlottHMM. Comparison with the findings using the STRAIGHT vocoder in Chapter 3 and Toshiba's pitch-synchronous Fourier transform (PSFT) vocoder in Chapter 4 allows for potential vocoder-specific findings to be identified. The third novel contribution of this chapter is investigating the effect of frequency loss within the modulation spectrum domain between natural vocoded trajectories and trajectories generated following MLPG. This is an interesting operating point as modulation spectrum scaling has recently gained popularity as a postfiltering method for 'undoing' effects of modelling speech (reinstating the modulation spectrum to the level in natural vocoded speech). The fourth contribution of this chapter is to include current enhancement methods into the 'continuum of speech' in order to test their effectiveness. Enhancement methods are created with the aim 'undoing' the effects of modelling on speech parameters generated from modelling. Therefore enhancement methods, within the conceptual 'continuum of speech', aim to move modelled speech back across the continuum towards natural speech. By placing these conditions into the continuum methodology, the extent to which enhancement methods successfully reverse effects of modelling can be investigated. By conducting this investigation at the same time as evaluating different contributing factors of hypothesised causes of

reduced quality in SPSS, perceptual comparisons as to the extent the enhancement methods are successful can be compared with respect to these causes of reduced quality.

5.2 Chapter overview

SPSS relies on the ability of the vocoder to decompose speech into a set of speech parameters that characterise the (perceptually) relevant aspects of speech. Most vocoders start from the source-filter representation (Fant, 1960), where a speech waveform is represented as a linear combination of an excitation signal and a resonant filter. In speech production, the corresponding components are the voice source and the vocal tract filter. In natural speech, the contributions of these components cannot be completely separated since they interact (Titze, 2008). The existence of this interaction between source and filter is well known, yet rarely taken into account in speech technology. It is possible that the failure to model this interaction between the two components might be one cause of poor quality in SPSS.

The aim of this chapter is to study the relative contributions of excitation and filter to the quality of synthetic speech. This includes an assessment of the degrading effect of vocoding and statistical modelling with regard to these two components. A focus of this investigation is the consequences of assuming that source and filter parameters are independent. Additionally, the effectiveness of three filter enhancement techniques is evaluated. In order to do this, the GlottHMM vocoder (Raitio et al., 2011b) is used in our experiments due to its ability to decompose speech into components corresponding closely to natural speech production: the glottal source signal and the vocal tract filter. A cross-synthesis scheme is adopted where speech is synthesised using the source and filter in all possible combinations of i) natural, ii) vocoded, and iii) modelled. The particular contributions of utterance-level variance of generated filter parameter trajectories and changes in modulation characteristics are assessed by applying both enhancing and degrading effects to the vocal filter parameters. All these combinations were assessed in two large subjective evaluations; one where listeners rated the speech samples according to similarity to each other, and another on a mean opinion score (MOS) scale. Note that, as in the investigations in Chapters 3 & 4, prosodic aspects were omitted from the investigations in this chapter in order to reduce the otherwise very large amount of testing required. Prosody is a large research field in its own right and as such is removed from testing in this thesis to allow better focus.

5.3 Vocoder

As noted above, the GlottHMM vocoder (Raitio et al., 2011b) is used in the experiments, primarily because GlottHMM aims to accurately model the two speech production components: the voice source signal and the vocal tract filter. Parameters are extracted at a fixed frame rate with a frame shift of 5ms. This vocoder is closer to natural speech production than conventional vocoders (e.g. STRAIGHT (Kawahara et al., 1999, 2001)). Such a vocoder may therefore prove beneficial in testing hypotheses concerning source and filter contributions and their interactions, with respect to SPSS quality. Another reason for choosing the GlottHMM vocoder is that it can be easily modified to accommodate the needs of this experiment; for example, it is straightforward to use a voice source signal derived from natural speech during synthesis. Finally since the STRAIGHT and PSFT vocoders have been investigated in Chapters 3 & 4, it will be interesting to perform a comparable investigation with a different type of vocoder.

GlottHMM is based on the conventional source-filter model, but the decomposition of speech into two components is based on the physiology of human speech production: the voice source signal and vocal tract filter. GlottHMM uses iterative adaptive inverse filtering (IAIF) (Alku, 1992), a glottal inverse filtering method based on all-pole modelling for that purpose. After the decomposition, the voice source signal is parameterised in detail, in order to enable accurate reconstruction of the signal in synthesis. The speech features used by the GlottHMM vocoder are shown in Table 5.1. Moreover, GlottHMM uses a natural glottal flow waveform as a base for creating the synthetic excitation in order to preserve the phase characteristics of the natural glottal flow. GlottHMM has been shown to produce synthetic speech which is both of high quality and very intelligible (Raitio et al., 2011b, Suni et al., 2010, 2011, 2012), and it has already been used in various experiments investigating voice source modelling in statistical speech synthesis (e.g. Raitio et al. (2011a, 2014, 2013)).

Table 5.1: *Speech features extracted by the GlottHMM vocoder.*

Feature	Order	Source	Filter
Frame energy (dB)	1	×	
Log-fundamental frequency	1	×	
Harmonic-to-noise ratio (dB)	5	×	
Voice source spectrum LSF	10	×	
Vocal tract spectrum LSF	30		×

5.4 Experiments

5.4.1 Speech material and voice building

A speech database of a British male speaker was used in the study in this chapter (Cooke et al., 2013a). The database consists of 2,022 read-aloud sentences selected for the purpose of speech synthesis, leading to approximately 2 hours of speech data (sampled at 16 kHz). The speech was parameterised using the GlottHMM vocoder, and an HMM-based voice was built following the standard HTS method (Zen et al., 2007a). Note that generation is without GV as variance restoration is kept as an enhancement method within the investigation. Delta and delta-delta features were appended to the speech features, and hidden semi Markov models (HSMMs) were used as acoustic models. All speech features, shown in Table 5.1, were modelled in individual streams except for the vocal tract spectrum LSFs and frame energy which were in the same stream.

5.4.2 Cross-synthesis methodology

In order to study the relative effect of the source and filter components at each processing stage of speech synthesis, a cross-synthesis scheme is used. In the cross-synthesis scheme three versions (natural, vocoded, modelled), of each of the two components are created, from which all possible permutations are used to synthesise speech. That is, stimuli were created which combined the properties of natural, vocoded and synthetic speech.

It is common in SPSS to apply some enhancement to some speech parameters following generation. The most common of these enhancement methods is global

variance (GV) (Toda and Tokuda, 2007). We applied four different enhancement or degradation methods to the filter parameter trajectories:

1. Global variance (GV) scaling. This method aims to replicate the performance of the GV enhancement described in Toda and Tokuda (2007) and is implemented in the same way as in Chapters 3 & 4. This method is implemented by subtracting the utterance-level mean for each coefficient from its respective coefficient trajectory. The resulting trajectory is then multiplied by a scalar value to increase or decrease the signal variance. The average coefficient-level variance values are calculated across the natural vocoded and HMM-generated trajectories for the training utterances. The scalar values used for the variance scaling conditions in this investigation are calculated so as to scale the coefficient-level variances towards the stored variance values for natural vocoded or HMM-generated trajectories, depending on whether it was being applied as an enhancement or simulation method. In practice this method is very similar to that described by Silén and Helander (2012), which was found to enhance the speech as much as GV.

2. Scaling of modulation spectrum (MS) (Takamichi et al., 2014b). The average utterance-level MS values were calculated across natural vocoded and HMM-generated LSFs for the training data sentences. This method then scaled the MS of LSF parameters towards the average MS values of natural or HMM-generated trajectories, depending on whether it was being applied as an enhancement or simulation method. This allows this MS scaling method to be applied to HMM-generated LSF parameters as a postfiltering method to restore the MS to that of natural vocoded speech (as it is usually used), or alternatively to natural vocoded LSF parameters as a simulation of the MS conditions in HMM-generated speech.

3. Temporal smoothing. This method is implemented as described in Chapters 3 & 4. A weighted moving average is calculated across the natural vocoded LSF signal in order to simulate temporally over-smooth trajectories following MLPG in SPSS.

4. Formant enhancement in the power spectrum domain, as described by Raitio et al. (2010). This method attempts to over-exaggerate the parameter trajectory before modelling by reducing energy of the spectrum in the low-energy regions. The aim is to account for the inevitable loss in temporal detail which will occur when modelling the parameter trajectory.

In this investigation one group of the stimuli starts from the natural vocoded filter and imposes certain properties of modelled speech by degrading the filter. The effect of statistical modelling is simulated by scaling the GV and MS of the LSF parameter trajectories to match the values seen in modelled speech, and by temporal smoothing. Another group starts from a modelled filter and applies enhancement procedures aiming at improving the quality (and thus moving the generated speech back across the 'continuum' towards natural speech). This is implemented by scaling the GV up to 50% of the way from modelled towards natural, scaling the MS up to 85% of the way towards natural (Takamichi et al., 2014b), and by applying formant enhancement to the LSFs in the power spectrum domain (Raitio et al., 2010). All the resulting 25 conditions resulting from the various possible combinations are shown in Table 5.2 and the filter processing techniques are shown in Table 5.3. The strengths of the filter processing techniques included in the listening test were selected following informal listening.

In order to have a reference 'perfect' source, the GlottHMM vocoder was used to extract the voice source signal for the natural source conditions, given the filter for each condition. In order to combine the source and filter parts of each combination, the statistically modelled features were generated using time-aligned labels. To make sure the alignment between natural source/filter and synthetic source/filter was as good as possible, the voiced and unvoiced regions of vocoded and modelled fundamental frequency parameters were compared, and only the best matching sentences were used in the experiments. Of the conditions included for testing, only *2-nat-voc* and *1-natural* have a perfect match between source and filter. In condition *4-voc-voc* the source is parametrised, at synthesis-time the source that will be used to excite the filter will not be the exact source that created the natural speech, instead it will be constructed from the parametrisation. Whereas for condition *2-nat-voc* the source is the exact residual once the estimated filter is removed from the speech signal. This means that in condition *2-nat-voc* the source and filter components match exactly.

5.4.3 Listening Tests

The perceptual testing for this investigation was in two phases, each employing a different paradigm: pairwise judgements analysed via multi-dimensional scaling (MDS), and mean opinion score (MOS) testing. The first of these involved listeners making "same or different quality" judgements about pairs of utterances generated under the

differing conditions in Table 5.2. From these responses a perceptual distance matrix can be constructed, from which MDS generates a visualisation which plots the listener responses in a fixed number of dimensions. The second paradigm, MOS testing, required listeners to rate single stimuli (the same set of utterances as in the previous test)

Table 5.2: *The 25 conditions investigated in the study, consisting of source and filter components from natural (nat), vocoded (voc), and modelled (hmm) speech. The filter processing methods are indicated in the last column (see definitions in Table 5.3).*

Condition name	Source	Filter	Filter processing
1-natural	nat	nat	
2-nat-voc	nat	voc	
2-nat-voc-ms−	nat	voc	MS–
2-nat-voc-smth	nat	voc	Smoothing
2-nat-voc-gv−	nat	voc	GV–
3-nat-hmm	nat	hmm	
3-nat-hmm-enh	nat	hmm	LSF-enh
3-nat-hmm-gv+	nat	hmm	GV+
3-nat-hmm-ms+	nat	hmm	MS+
4-voc-voc	voc	voc	
4-voc-voc-ms−	voc	voc	MS–
4-voc-voc-smth	voc	voc	Smoothing
4-voc-voc-gv−	voc	voc	GV–
5-voc-hmm	voc	hmm	
5-voc-hmm-enh	voc	hmm	LSF-enh
5-voc-hmm-gv+	voc	hmm	GV+
5-voc-hmm-ms+	voc	hmm	MS+
6-hmm-voc	hmm	voc	
6-hmm-voc-ms−	hmm	voc	MS–
6-hmm-voc-smth	hmm	voc	Smoothing
6-hmm-voc-gv−	hmm	voc	GV–
7-hmm-hmm	hmm	hmm	
7-hmm-hmm-enh	hmm	hmm	LSF-enh
7-hmm-hmm-gv+	hmm	hmm	GV+
7-hmm-hmm-ms+	hmm	hmm	MS+

on a 5 point scale between 'bad' and 'excellent' in terms of the quality of the speech. Whilst MDS is potentially quite powerful, it can sometimes be difficult to draw precise conclusions from the complex plots it produces; the MOS test was included to provide a basis for the interpretation of the MDS analysis.

5.4.3.1 Pairwise listening test

In the first listening test, listeners were presented with pairs of stimuli in which every condition was paired with every other condition (but not itself). Each possible pair of 25 conditions was repeated 12 times resulting in 7200 pairs. The pairs were presented in a randomised order to minimise bias. The sentence (i.e., text) was different for each utterance within a pair, with the sentences being drawn otherwise at random from a set of 40 sentences. The presentation order of the pairs was such that no sequence of two pairs contained the same sentence more than once, but was otherwise random. 45 native English-speaking participants with no known hearing impairments were recruited for this test. The 7200 pairwise comparisons were divided amongst the participants, with each participant making 160 pairwise quality judgements. This number of judgements has previously been demonstrated to be reasonable for subjects Mayo et al. (2011).

Table 5.3: *The symbols and explanations for the processing methods applied to the filter parameter trajectories.*

Symbol	Explanation
GV– (for vocoded)	Global variance (Toda and Tokuda, 2007) scaled down to the level of synthetic speech
GV+ (for hmm)	Global variance (Toda and Tokuda, 2007) scaled up by 0.5 towards the level of natural speech
MS– (for vocoded)	Modulation spectrum (Takamichi et al., 2014b) scaled down to the level of synthetic speech
MS+ (for hmm)	Modulation spectrum (Takamichi et al., 2014b) scaled up by 0.85 towards the level of natural speech
Smoothing (for vocoded)	Smoothing the trajectory with a Hann window of length 21
LSF-enh (for hmm)	Formant enhancement applied to LSFs in the power spectral domain (Raitio et al., 2010)

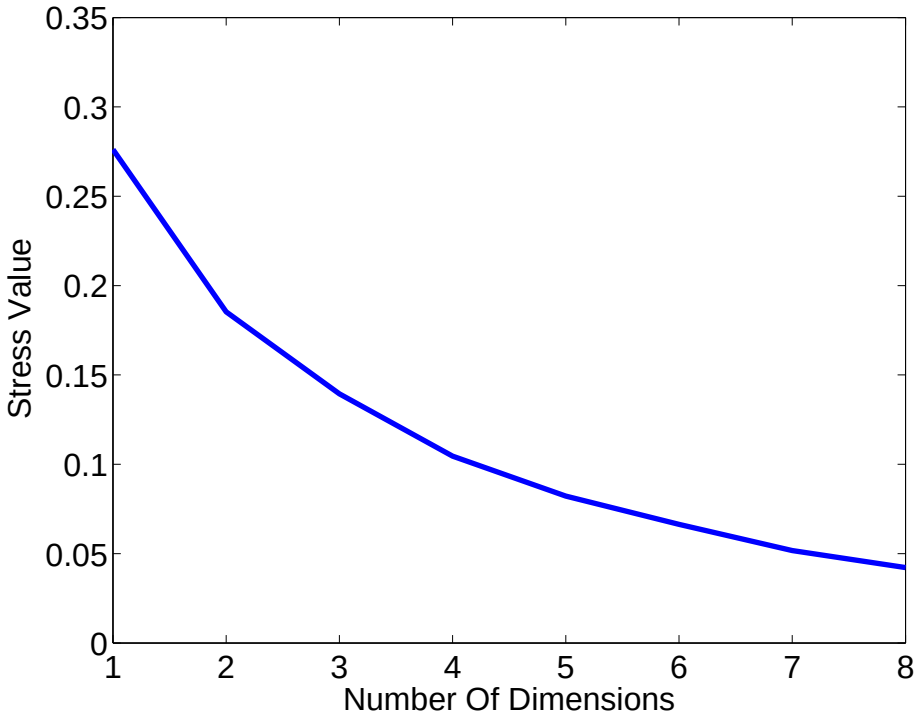


Figure 5.1: *Stress levels returned by MDS at different dimensions*

5.4.3.2 Single stimulus listening test

In the MOS test, each of the testing conditions was presented to each listener 4 times, for randomly selected utterances. Thus, each listener evaluated 100 samples. The presentation order was such that no sequence of two utterances involved either the same sentence or the same condition more than once, but was otherwise random. The same 40 test sentences from Section 5.4.3.1 were used in this test. 20 native English-speaking participants with no known hearing impairments were recruited for this test.

5.5 Results

5.5.1 MDS plot

As described in Chapter 1, the listener responses from the "same or different quality" task provides a perceptual distance matrix between the conditions tested, which is then projected into a fixed number of dimensions using MDS. The stress values across the listener responses at various dimensions is shown in Figure 5.1. Based on the principles for selecting an appropriate number of dimensions to visualise listener responses, described in Chapter 1, two dimensions were selected as a reasonable operating point.

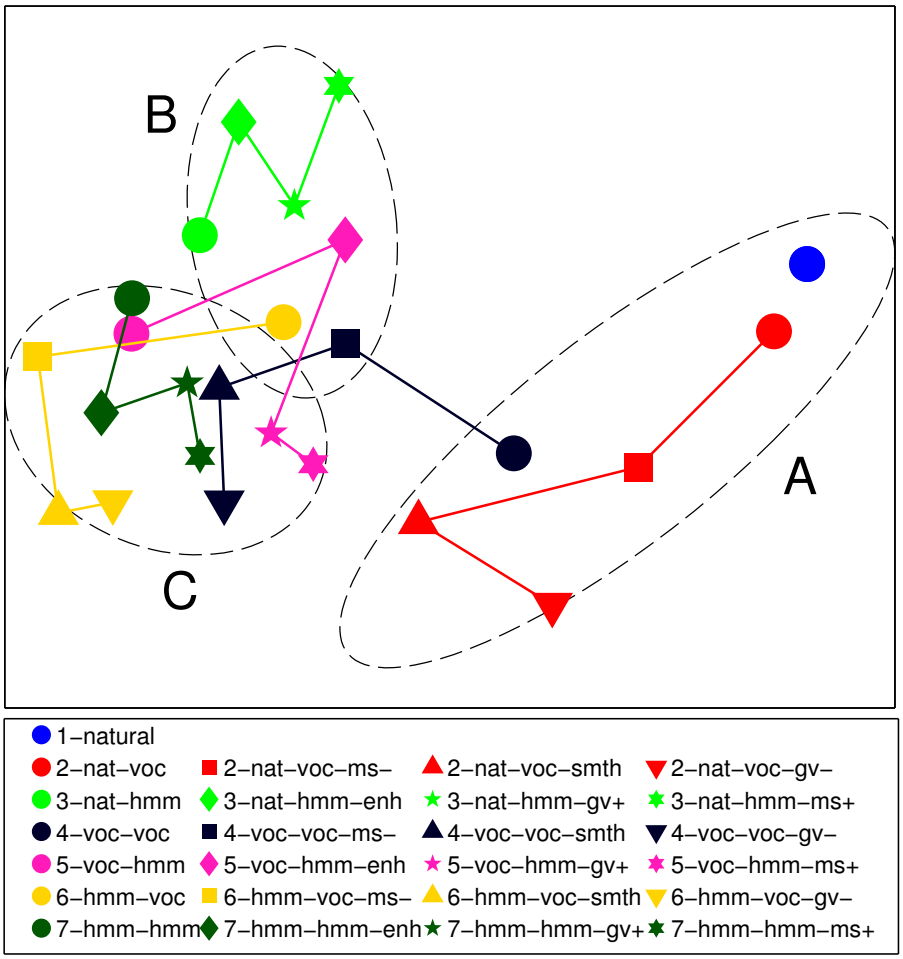


Figure 5.2: *MDS plot for 2 dimensions. The dashed ellipsis show the clustering of the conditions in the MDS space, as reached in 80% of cases by performing k-means clustering. Figure appeared in Merritt et al. (2014).*

In the visualisation plot, there is one point for each condition. Those conditions which listeners judged to be more perceptually similar will be closer together in the plot. Figure 5.2 shows the MDS plot at 2 dimensions. These findings will now be discussed.

5.5.1.1 Voice source

The points (i.e., conditions in Table 5.2) cluster into 3 groups:

A) Natural speech (*1-natural*), all systems with natural source and perfectly matched filter (*2-nat-voc-x*), and pure vocoded speech (*4-voc-voc*).

B) All systems with natural source and modelled filter (*3-nat-hmm-x*), vocoded source and modelled filter with formant enhancement (*5-voc-hmm-enh*), modelled source

and vocoded filter (*6-hmm-voc*), and vocoded source and filter with decreased modulation spectrum (*4-voc-voc-ms−*).

C) All other conditions.

The clustering was first performed by eye, and then confirmed by k-means clustering, with the depicted clustering being the most common outcome (on 80% of occasions). The clustering shows that using the natural source is the biggest single factor, indicating that a better source signal has the potential to substantially improve the quality of the resulting speech. However, the clustering also tells us that any mismatch between source and filter has damaging perceptual consequences.

Vocoded and modelled voice sources (*5-voc-hmm* and *7-hmm-hmm*) are very close to one another when using a modelled filter, indicating that the modelling of the source is not a restricting factor in this situation. However, the filter enhancements are slightly more effective in the case of vocoded source than when combined with the modelled source.

5.5.1.2 Vocal tract filter

Using a vocoded source in combination with a vocoder filter (*4-voc-voc*) or modelled filter (*5-voc-hmm*) are perceptually very similar in the cases when the vocoder filter is degraded, and the modelled filter is enhanced. This suggests that these enhancements are working when applied to modelled filters, although they do not quite restore the speech to the quality of vocoded speech: listeners can still easily distinguish them.

The perceptual closeness of conditions with vocoded source and modelled filter (*5-voc-hmm*) and HMM synthesis (*7-hmm-hmm*) would strongly suggest that the current quality of SPSS systems (that make a source/filter independence assumption), is limited mainly by the modelling of the filter. The *6-hmm-voc* condition is closer to natural speech than either of these two conditions, further supporting this conclusion.

5.5.1.3 Source and filter interaction

The perceptual distance between the conditions with HMM source and vocoded filter (*6-hmm-voc*) and the HMM-synthesis (*7-hmm-hmm*) are generally small, once degradation and enhancement effects are applied respectively. This is interesting: applying the enhancement to HMM-synthesis appears *not* to make the speech quality noticeably different, in contrast to vocoded source and modelled filter (*5-voc-hmm*). This could

indicate either: 1) there is something natural about the vocoded source that modelling fails to capture; or 2) once the source and filter have been independently modelled, very little can be done to recover from that.

The large perceptual distance between systems with natural source and vocoded filter (*2-nat-voc*) and natural source and modelled filter (*3-nat-hmm*) (among the largest between system configurations) can be interpreted in two ways: 1) it could be caused by artefacts introduced by mismatches between source and filter in *3-nat-hmm*; or 2) it could be due to the differences between vocoded and modelled filter coefficients when excited by the 'perfect' natural source, resulting in a match between source and filter in one condition and not in the other.

Natural source with vocoded filter conditions using MS degradation (*2-nat-voc-ms−*) and smoothing (*2-nat-voc-smth*) both lie perceptually close to vocoded speech (*4-voc-voc*). This may indicate that MS down-scaling and smoothing both introduce mismatch between the source and filter, similar to the effect introduced by vocoding using the current source-filter representation of speech production.

5.5.2 MOS scores

The results of the MOS test are shown in Fig. 5.3. These largely back up the main conclusions from the MDS analysis: that increase in distance from the natural speech point in the MDS plot corresponds closely to decrease in speech quality, and that the filter enhancements applied to the HMM filter produce noticeable improvements in the quality of speech. The results of significance testing of the listener responses from the MOS test is shown in Figure 5.4, using the t-test and Wilcoxon signed-rank test at a p value of 0.05. These significance tests are described in Chapter 4. Holm-Bonferroni correction was applied due to the large number of condition pairs to compare.

An interesting contradiction between the results from the MDS and MOS tests is the scores for the natural source with modelled filter configuration (*3-nat-hmm*). In the MDS plot, the conditions closest to natural speech were GV and MS up-scaling (GV+ and MS+). However the results from the opinion score test showed that listeners prefer the speech with formant enhancement (LSF-enh) and GV up-scaling (GV+). This shows that listeners in the MDS test were not simply making one-dimensional preference comparisons and were instead making their judgements along more than one dimension of difference. Other points of interest from these results will now be discussed.

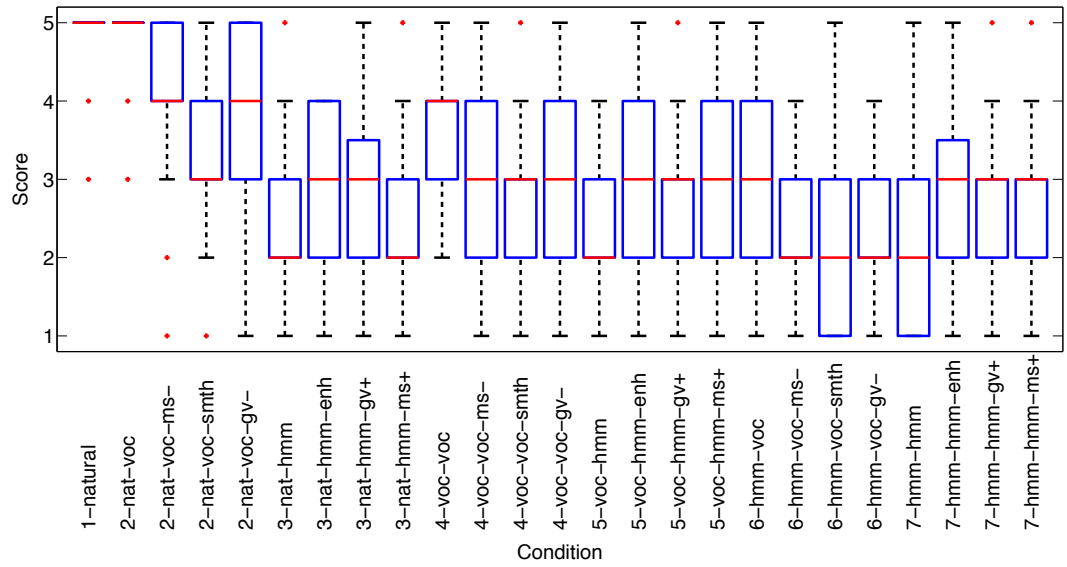


Figure 5.3: *Box plot of listener opinion scores. Plot uses the same notation as in Figure 4.2. Figure appeared in Merritt et al. (2014).*

5.5.2.1 Voice source

Conditions *2-nat-voc-ms−* and *4-voc-voc* receive similar quality scores, which coincides with the findings of the MDS test that these are perceptually similar. However *2-nat-voc-smth* is not rated as highly and instead there is a preference for *2-nat-voc-gv−*, which was the furthest point in the *2-nat-voc* system configuration in the MDS plot, highlighting that speech produced under this set of conditions is high in quality.

5.5.2.2 Vocal tract filter

The MOS results for vocoded speech (*4-voc-voc*) and for vocoded source and modelled filter (*5-voc-hmm*) support the findings of the MDS test, in that speech under these conditions, following degradations and enhancements respectively, have very similar quality. This supports the observation that the effects caused by statistical modelling are being perceptually repaired to some extent, but the speech is still of noticeably worse quality than vocoded speech.

The modelling of the filter parameters in *5-voc-hmm* may be a key factor limiting the quality output when source and filter are determined independently, as the quality score of *5-voc-hmm* and *7-hmm-hmm* remain similar whereas *6-hmm-voc* is rated better in quality by listeners. However this test found little difference between *6-hmm-voc* and the *5-voc-hmm* configurations once filter enhancements are applied.

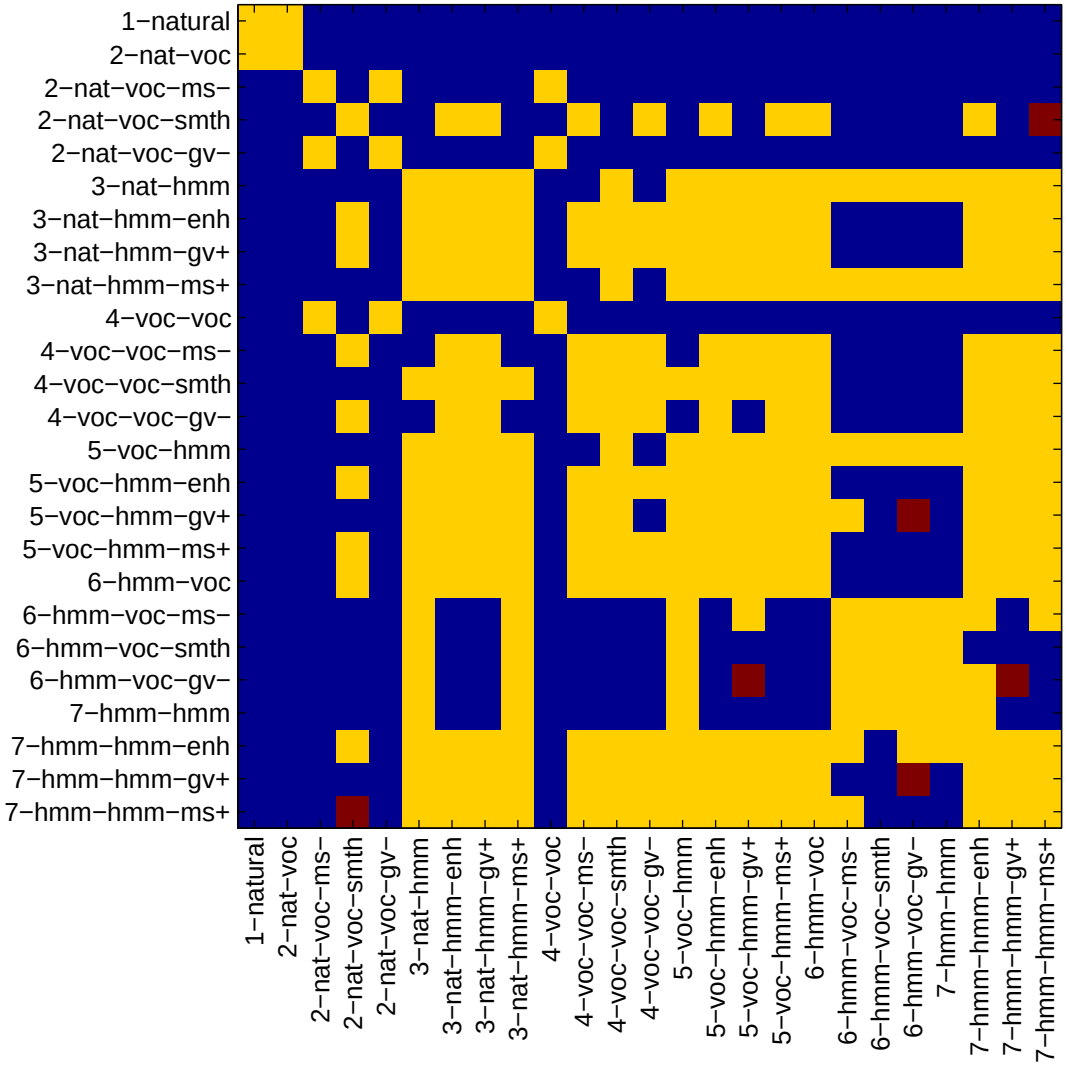


Figure 5.4: *Visualisation of significant differences between systems in terms of absolute value using t-test and the Wilcoxon signed-rank test (p=0.05). Dark blue indicates agreement in significant difference. Yellow indicates agreement in no significant difference. Red indicates significant difference found using t-test but not with Wilcoxon signed-rank test.*

5.5.2.3 Source and filter interaction

The results for natural source and vocoded filter show the largest quality drop when using smoothing (*2-nat-voc-smth*). Smoothing of speech trajectories may create the largest decrease in the interaction, or degree of consistency, between the source and filter, by averaging content across consecutive frames and removing all fine variations from the trajectories. MS and GV degradation have less effect than smoothing, presumably because they are preserving more of the source-filter interaction, in other words, that the frame-by-frame variations in the filter parameters are consistent with the frame-by-frame variations in the source.

Applying enhancements to HMM-based speech (*7-hmm-hmm*) does not help as much as when they are applied to the condition using a vocoded source and HMM filter (*5-voc-hmm*). Possible explanations were already offered for this in Section 5.5.1.

5.6 Conclusion

The methodology introduced in Chapter 3 has been further extended to investigate the effects introduced by the modelling of source and filter coefficients, along with investigating the effectiveness of three filter enhancement techniques. By creating appropriate stimuli, performing two listening tests, and analysing the results, it has been possible to see clear differences in quality as source and/or filter are varied from natural, through vocoded to HMM-generated.

Current filter enhancement techniques are able to recover some of the quality loss caused by modelling the filter, yet the final quality seems to be more affected by the interaction of source and filter than by the individual quality of either one alone. Whilst it is impossible to 'prove' anything beyond reasonable doubt using perceptual tests, our results provide supporting evidence that the assumption of independence between source and filter, which is inherent in standard statistical parametric speech synthesisers, is one of the most significant limiting factors on the quality of synthetic speech. The effect of this independence assumption will be further tested in Chapter 6.

Retrospectively it is apparent that the MOS test conducted in this chapter is not perfect. Utterances and conditions were selected at random, independent of each other. Whilst each condition was rated 80 times in total, there was no balancing of condition with respect to utterance. This means that not all utterances were presented under all conditions, although in practice each condition was presented under at least 32 dif-

ferent utterances. This test design also results in an imbalance between the number of times the different utterances, under each condition, were presented, potentially resulting in a slight skew of responses towards specific utterances. Additionally conditions were not balanced by listener, meaning that each listener was not guaranteed to hear each condition exactly twice. Although this test design is not ideal, it is still believed that the responses from this test are representative of the quality of the various conditions tested. This is because there are a large number of judgements made for each condition (80), over a large number of different utterances (between 32 and 38) and presented in the constrained manner described in Section 5.4.3.2. Also, these responses are in line with those from the 'same or different' task.

Chapter 3 introduced a methodology for investigating hypothesised causes of reduced quality in HMM speech synthesis which has been used in Chapters 3 & 4, as well as in this chapter, to further our understanding. Chapter 6 will introduce an alternative methodology for investigating the effects of modelling assumptions applied in SPSS, using a specially crafted corpus of speech comprising of utterances recorded numerous times.

Chapter 6

Measuring the perceptual effects of modelling assumptions in speech synthesis using stimuli constructed from repeated natural speech

The investigation in this chapter was conducted by a group of researchers, including myself, and was previously published in Henter et al. (2014a). The sentences were selected for the REHASP 0.5 corpus by myself from the Harvard sentences in order to select those most suitable for a British English speaker. The algorithm used to create the constrained random ordering of sentences to be read by the speaker for the REHASP 0.5 corpus was created by myself. The recording of the speaker was performed by myself and Gustav Eje Henter. The discussions over the configurations to be included in this investigation was done by myself, Gustav Eje Henter and Matt Shannon. The configurations used for this investigation were coded by Matt Shannon. The multi-dimensional testing was designed and results interpreted by myself. The MUSHRA test was designed and results interpreted by Gustav Eje Henter. The published version of the paper was predominantly written by Gustav Eje Henter; this chapter has been rewritten in my own words.

6.1 Introduction

In Chapter 3, a methodology for investigating hypothesised causes of reduced quality of speech synthesis from SPSS systems was introduced. This methodology uses a

conceptual 'continuum of speech', ranging from natural speech at one end of the continuum to full HMM-generated speech at the other. Each of the hypothesised causes of reduced quality within SPSS are elements within this continuum. This allows for independent perceptual testing of each hypothesised cause rather than attempting to piece apart contributing factors from HMM-generated speech, where the contributing causes of reduced quality are co-occurring. Chapters 4 & 5 extended this methodology to investigate a variety of hypothesised causes of reduced quality in SPSS.

The investigations in Chapters 3, 4 & 5 tested hypothesised causes by applying simulations to natural vocoded speech parameters, applying constraints to HMM-generated speech or by applying enhancement methods to parameters generated from a HMM synthesis system. This chapter will investigate further causes of reduced quality of SPSS systems within the conceptual 'continuum of speech'. It focuses on the quality achievable when common modelling assumptions in SPSS systems are enforced. This is investigated through the use of a specially crafted speech corpus of speech which is composed of repeated examples of utterances (Henter et al., 2014b). This corpus is used to estimate the upper-bound performance of statistical parametric speech synthesis (SPSS) when various commonly applied modelling assumptions are in place, if we were able to perfectly generate speech from the models.

6.2 Assumptions in SPSS

There are a wide range of modelling assumptions made in SPSS systems. This chapter will focus on a few of these.

- The first of the assumptions to be investigated is that the process of vocoding has no detrimental effect on quality. By this, it is meant that the process of parametrising speech into a representation suitable for modelling, before then transforming this parametrisation back into the time-domain waveform, is a fully reversible process (i.e., doesn't have adverse effects on the quality of speech). Chapters 4 & 5 have already indicated that this assumption appears to not be the case.

- An additional assumption present in standard SPSS systems is that duration is independent of the HMM model, given that HSMMs are typically used to determine state-wise durations. This results in a Gaussian distribution being used to determine the duration rather than the exponential distribution given by the

HMM transition probability. Duration is usually calculated prior to generation of features from the SPSS system and in HMM synthesis the predicted duration usually has no effect on the model selected.

- The independence of source and filter aspects of speech production is another assumption present in many aspects of SPSS. Chapter 5 discussed that many vocoders are based on a source-filter representation of speech, assuming that the contribution of these two elements of speech production are independent; however, they are known to interact. The assumption of independence between source and filter parameters is realised in SPSS by these features usually being modelled using separate decision trees. This results in potentially different clustering of source and filter parameters, meaning different linguistic contexts (and therefore different frame-wise parameters) are clustered together to produce models for the source and filter components. This potentially introduces a large mismatch between generated source and filter parameters. An extension of the source-filter independence assumption is the assumption of speech parameter stream independence. There is typically more than one stream representing the source component in SPSS systems (e.g., f_0 & BAP). It is standard practice in HMM synthesis to cluster each of these parameter streams with a separate decision tree, potentially further increasing the mismatch between parameter streams.

- HMM synthesis models typically use diagonal covariance, meaning that there is an assumption of independence across the coefficient trajectories within the filter parametrisation (this is usually Mel-cepstra parameters). This independence assumption has been investigated previously in the literature, with the introduction of semi-tied covariance matrices (Gales, 1999), however these are still not commonplace in SPSS systems.

- Finally, speech parameter models are constructed assuming a Gaussian distribution of speech data. It is then assumed that meaningful model parameters will be produced by performing averaging of the samples present in the leaf node following decision tree clustering. This means that, by averaging these samples together, the mean value provides a parameter value which represents the target linguistic context well. It is entirely possible that the mean value instead is situated between values that were realised in the data and was never actually a realised value itself.

In SPSS, once the parametric models have been learnt there is still the issue of how to generate parameters from these (as mentioned in Chapter 1). The common approach is to use the mean value of the Gaussian distribution fitted during training. Maximum likelihood parameter generation (MLPG) then combines the Gaussian mean value, which is constant across the state, with dynamic features and variances to produce parameter trajectories. However this form of generation favours the mean value of the distribution, which as stated above may not be a good data point. This results in generating trajectories with reduced global variance (GV) and is why GV postfiltering is required in SPSS. An alternative is to perform sampling from the model (Shannon et al., 2011). When sampling from the model each of the different independence assumptions has a cumulative effect on the generated speech parameters. This means that parameters generated under a model which assumes duration independence produces different parameters than a model which assumes independence between source and filter and independence between Mel-cepstra coefficients. However when the mean of the distribution is used instead of performing sampling (as is more standard in SPSS parameter generation) to generate vocal tract filter parameters, the generated parameters are the same for each of these different models of speech.

This investigation aims to measure the effect that these different modelling assumptions have on the naturalness of synthesised speech produced, under upper-bound conditions, assuming perfect generation from the models is possible. There are many slight variations in how a single utterance can be read, even when read within the same style (e.g., neutral). Each repetition of the same utterance under the same reading style represents a completely natural realisation of the utterance by a speaker. As such, if we had a hypothetical 'perfect' model combined with 'perfect' parameter generation (both sampling and mean-based generation methods are investigated in this chapter), each of the different realisations of an utterance represents perfectly natural 'generated' speech. By substituting different realisations of the same utterance into different parameter streams in the output speech, we are able to simulate upper-bound performance under different independence assumptions. This chapter aims to investigate the implications on synthesis performance, of our hypothetical 'perfect' system, as a consequence of the discussed assumptions currently in place in SPSS systems. To do this, multiple repetitions of natural realisations of the same utterance will be combined to demonstrate the effect of each of these assumptions. With each of these repetitions representing a completely natural realisation of the utterance, if these assumptions are harmless then there should be no consequence for naturalness.

6.3 The REHASP 0.5 corpus

In order to implement our idealised 'perfect' modelling and generation of speech parameters, under the modelling assumptions to be tested, a specially-crafted corpus is required. This corpus comprises repeated readings of utterances by a single speaker. Each recording is referred to as a repetition of an utterance. The utterances were from the Harvard sentences. The Harvard sentences are widely used within the speech technology field and are roughly phonetically balanced within each set of ten utterances. Three sets of these utterances (30 sentences) were selected for inclusion with each of the sentences being repeated 40 times. The selection of sentences to be included in the corpus was made such that these utterances should be suitable for a British English speaker.

A female British English speaker, "Lucy", was recorded in a hemi-anechoic chamber. The speaker was instructed to speak in a neutral style and to not intentionally vary the repetitions of each utterance. In order to avoid list effects, utterances were presented in a randomised order, with the exception that the same utterance was not allowed to appear twice in a row within the recording script.

The corpus therefore consists of 1200 recordings, originally being recorded at 96 kHz, 16 bit. Recordings were end-pointed to keep (at most) 100 ms of silence before first voice activity and 300 ms of silence after last voice activity, removing excessive silence from the speech data. A high-pass filter was applied to the audio files to remove minor low frequency electrical interference. The recordings were then downsampled to 16 kHz and amplitude was normalised to -24 dBov (ITU Recommendation ITU-T P.56, 2011). Both the original unprocessed recordings and the downsampled and processed recordings are freely available as the REHASP 0.5 corpus (Henter et al., 2014b). The downsampled and processed versions of the recordings are used in the remainder of this chapter.

6.4 Methodology

The STRAIGHT vocoder was used to extract speech features in this investigation. The speech features which make up the 'vocoded' speech used in this investigation are: 40 order Mel-cepstra coefficients, log-f_0 and 5 BAPs. These features were calculated with a frame shift of 5 ms. Temporal smoothing (by passing a sliding Gaussian window across the trajectories independently, where $\sigma = 0.8$ frames) was applied to the natural

vocoded parameters as informal listening suggested this removed artefacts from the speech. The remainder of the conditions included in this investigation operate from the condition which performs duration adjustment of utterances (condition D), representing the duration independence assumption within this investigation, as discussed in Section 6.2. This condition is implemented via dynamic time warping (DTW) in order to align different spoken realisations of the same text. The duration normalisation via DTW involves repetitions and deletions of frames from a recorded repetition of an utterance, in order to time-align the repetition to a reference duration, taken from a seperate repetition of the same utterance. DTW computes the alignment between the frames in the source and reference repetitions by minimising the Mel cepstral distortion (MCD), excluding the 0^{th} cepstral coefficient. Once aligned, the utterance repetitions can be used to represent the idealised modelling and generation of speech parameters under different modelling assumptions.

To represent the upper-bound performance in a system using a source-filter independence assumption, where these two elements of speech production are modelled separately, the parameterisation of the source element of speech (log-f_0 and BAPs) is taken from one repetition of the utterance whereas the filter representation (Mel-cepstra) is taken from a separate repetition. As mentioned above, these repetitions are time-aligned to a 'reference' duration from a separate repetition of the same utterance. The subsequent speech represents upper-bound modelling performance using an idealised generation approach which samples from this model. This approach can be further extended to investigate where all speech parameter streams (log-f_0, BAPs and Mel-cepstra) are modelled independently, with each stream coming from a separate, time-aligned, repetition. Diagonal covariance can be investigated by taking different Mel-cepstra coefficients from separate time-aligned repetitions of the utterance. In this investigation the effect of diagonal covariance is investigated at differing levels, observing its effect on different coefficient blocks (lower coefficient values and higher coefficient values). Finally the effect of averaging vocal tract filter parameters is investigated by time-aligning all repetitions of the test utterance, and taking the frame-wise mean value across the repetitions to represent the vocal filter parameter to synthesise. Table 6.1 shows the full range of conditions tested.

Table 6.1: *Conditions included in listening test. Table shows model configuration, generation method and the construction of the condition using examples from the REHASP corpus. The letters a, b, c and d indicate separate repetitions of an utterance. An asterisk indicates all coefficients come from separate repetitions. $\bar{x}$ indicates that an average over all repetitions was used.*

Condition			Parameter trajectory sources					
			Dur-	Source		Filter (MCEPs)		
ID	Description	Generation	ation	log-f_0	BAP	0-5	6-12	13-39
N	Natural speech	-	-	-	-	-	-	-
VU	Vocoded (unsmoothed parameters)	-	a	a	a	a	a	a
V	Vocoded (smoothed parameters)	-	a	a	a	a	a	a
D	Time-warped to reference duration	Sampling	b	a	a	a	a	a
SF	Source and filter independent	Sampling	b	a	a	c	c	c
SI	All parameter streams independent	Sampling	b	a	d	c	c	c
L1	Lower 6 MCEPs independent	Sampling	b	a	a	*	c	c
L2	Lower 13 MCEPs independent	Sampling	b	a	a	*	*	c
H1	MCEPs above 12 independent	Sampling	b	a	a	c	c	*
H2	MCEPs above 5 independent	Sampling	b	a	a	c	*	*
I	All MCEPs independent	Sampling	b	a	a	*	*	*
M	MCEPs averaged	Mean	b	a	a	$\bar{x}$	$\bar{x}$	$\bar{x}$

6.5 Experiments

The effects of the different modelling assumptions included in this investigation were tested under two different subjective tasks: a MUSHRA test, measuring naturalness ratings, and a 'same or different naturalness' comparison test, similar to the tests run in Chapters 3, 4 & 5. As described earlier, these conditions represent the upper-bound synthesis performance possible under different modelling assumptions. This upper-bound performance relates to otherwise 'idealised' modelling and generation of speech parameters while the modelling assumptions are enforced. The different conditions are created using our novel investigation paradigm, as described in Section 6.4, using the REHASP 0.5 corpus.

6.5.1 MUSHRA test

The full range of conditions, as shown in Table 6.1, were presented in a MUSHRA listening test (described in Chapter 1). Natural speech is provided as a hidden (listeners are not informed as to which of the condition sliders represents this condition) upper-anchor for listeners' judgements. Listeners are informed that this natural speech condition should be scored as 100 (completely natural). Lower-anchoring was not used in this test for synthesis evaluation for the reasons already mentioned in Chapter 1.

All 30 sentences in the REHASP 0.5 corpus were used for testing. The sentences were divided into the 3 sets of 10 phonetically balanced sets of Harvard sentences. 30 native English participants with no known hearing impairments were recruited. Each listener rated 20 screens (2 of the 3 sets of sentences), where each screen presents all 12 conditions. This resulted in each of the 3 sets of Harvard sentences being rated by 20 listeners. One subject didn't fully complete the MUSHRA test (they completed 15 out of their allotted 20 screens), resulting in a total of 595 sets of MUSHRA results.

6.5.2 Pairwise discrimination test

For the pairwise discrimination task, listeners were presented with pairs of conditions and asked whether the conditions are the 'same or different' in terms of naturalness. From these responses we are able to build up a matrix of perceptual 'difference' between each of the conditions. No comparisons were made between matching conditions in order to reduce the number of required comparisons to be made. Given the 12 conditions included in the listening test, this results in 132 unique condition comparisons. Each comparison is made 20 times, resulting in a total of 2640 comparisons being made across all listeners. The pairings of conditions to be compared, from Table 6.1, were presented in a randomised order. The utterances under which these conditions were compared, was also selected randomly. The 6 stimuli within any 3 consecutive comparison pairs were constrained to consist of 6 unique sentences. 20 native English speakers with no known hearing impairments were recruited to participate in the listening test. Each listener performed 132 comparisons. The matrix of perceptual 'differences' compiled across all listeners is then transformed into a fixed dimensional representation using multidimensional scaling (MDS).

6.6 Results

6.6.1 MUSHRA test

The responses from the MUSHRA test in terms of the absolute values of the scores awarded are shown in Figure 6.1. All tests for significant differences between conditions applied Holm-Bonferroni correction due to the large number of condition pairs to compare. All conditions are significantly different from all others in terms of absolute rating, except for between: SF and SI, SI and M, H1 and L1, H2 and L2. Significant differences are in agreement using a t-test and Wilcoxon signed-rank test at a p value of 0.01. These significance tests are described in Chapter 4. The agreement between the t-test and Wilcoxon signed-rank test, in terms of significant differences found, is illustrated in Figure 6.2.

The responses from the MUSHRA test in terms of the rank order given to the conditions are shown in Figure 6.3. All tests for significant differences between conditions applied Holm-Bonferroni correction due to the large number of condition pairs to compare. All conditions are found to be significantly different from all others in terms of the rank order, except for between: SF and SI, H1 and L1. Significant differences are

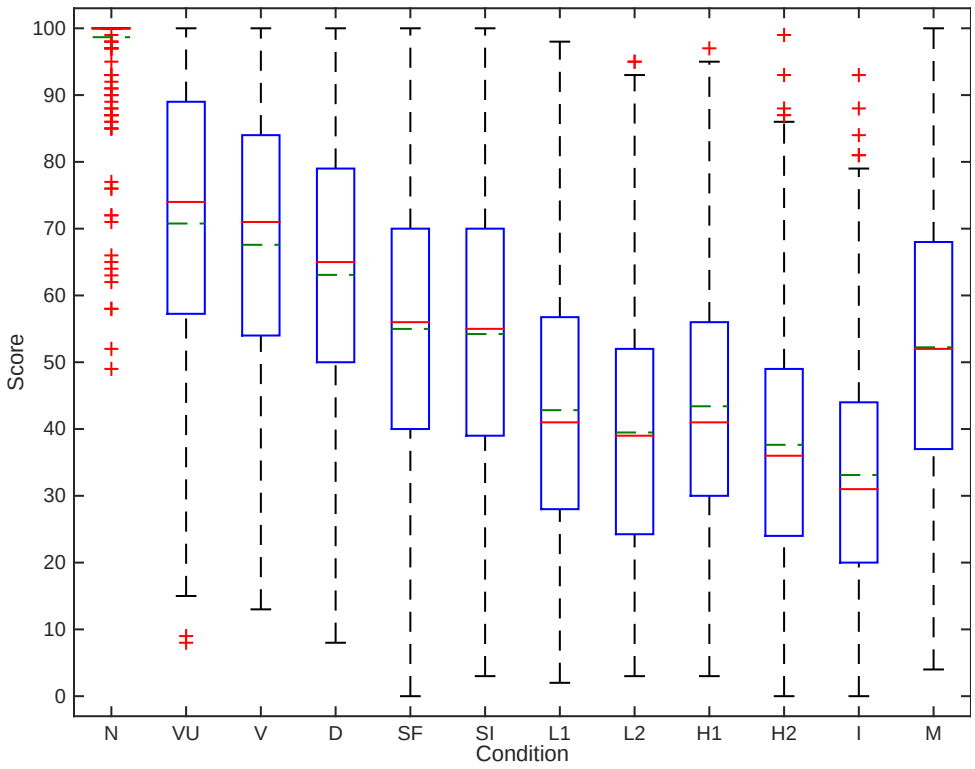


Figure 6.1: *Boxplot of absolute values from MUSHRA test. Plot uses the same notation as Figure 4.1.*

in agreement using the Mann-Whitney U test and the Wilcoxon signed-rank test at a p value of 0.01. The Mann-Whitney U test (also known as the Wilcoxon rank sum test) was selected to identify significant differences as this test handles data in terms of rank order rather than absolute differences, suiting the data being tested. This test is applied to cumulated results rather than matching samples, as with the Wilcoxon signed-rank test. The Mann-Whitney U test identifies significant differences between the distributions of the two classes being tested, however does not assume the two classes are normally distributed. The agreement between the Mann-Whitney U test and the Wilcoxon signed-rank test, in terms of significant differences found, is illustrated in Figure 6.4.

The responses from the MUSHRA test appear to indicate four groups of conditions, based on the fundamental assumptions applied: natural speech (N), vocoded speech (VU, V and D), stream independence (SF and SI) and diagonal covariance across Mel-cepstra coefficients (L1, L2, H1, H2 and I). An interpretation of these results is that a large drop of naturalness occurs as a result of vocoding alone, as was found to be the case in Chapters 4 & 5. Next, the assumption that source and filter parameters are suitable for independent modelling results in a further drop in naturalness from con-

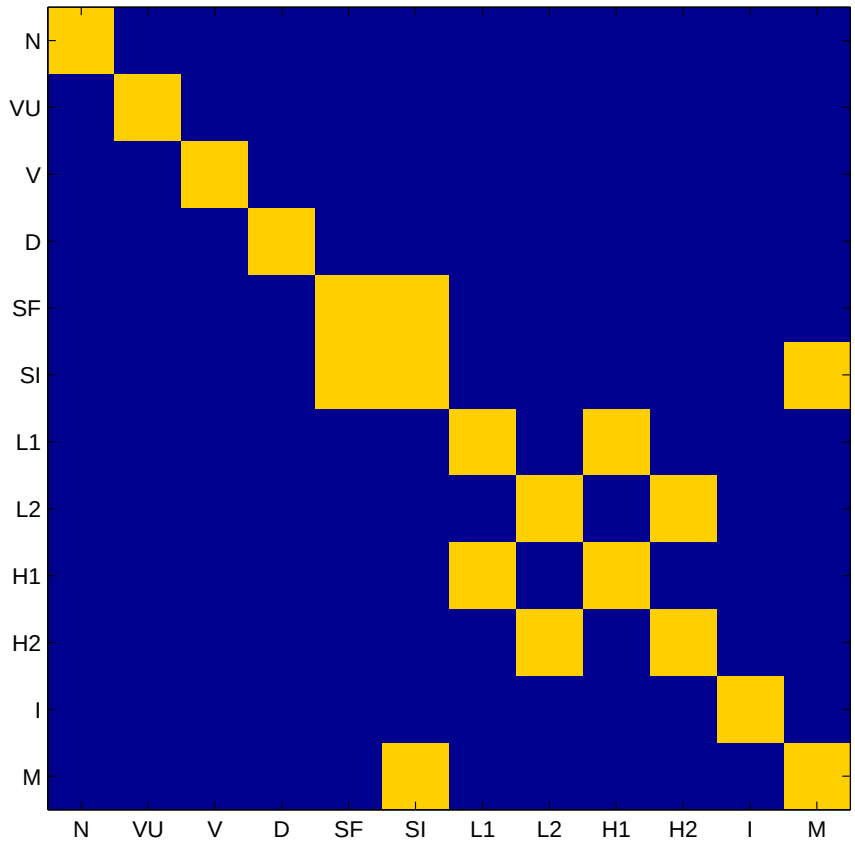


Figure 6.2: *Visualisation of significant differences between systems in terms of absolute value using t-test and the Wilcoxon signed-rank test (p=0.01). Dark blue indicates agreement in significant difference. Yellow indicates agreement in no significant difference.*

dition D (which is used to align the different repetitions of the utterance) to condition SF. Finally, assuming diagonal covariance across Mel-cepstra coefficients results in a large drop in naturalness, from condition SF to conditions L1, L2, H1, H2 and I. This is found to be the case across higher order coefficients as well as lower order coefficients. These findings indicate that the source-filter independence assumption and the diagonal covariance assumption are both inadequate for generating natural speech when sampling from the model. Also of interest from these findings is that where the mean value of the vocoded parameters is used (condition M) the naturalness falls into the 'category' of stream independence. Note that condition M is the same regardless of the underlying model assumptions in place (conditions D, SF, L1, L2, H1, H2 or I). This would appear to indicate that using the mean, rather than sampling, from the 'ideal' model of speech has different effects under different modelling assumptions made. Firstly, using the mean rather than sampling from the model provides gains in

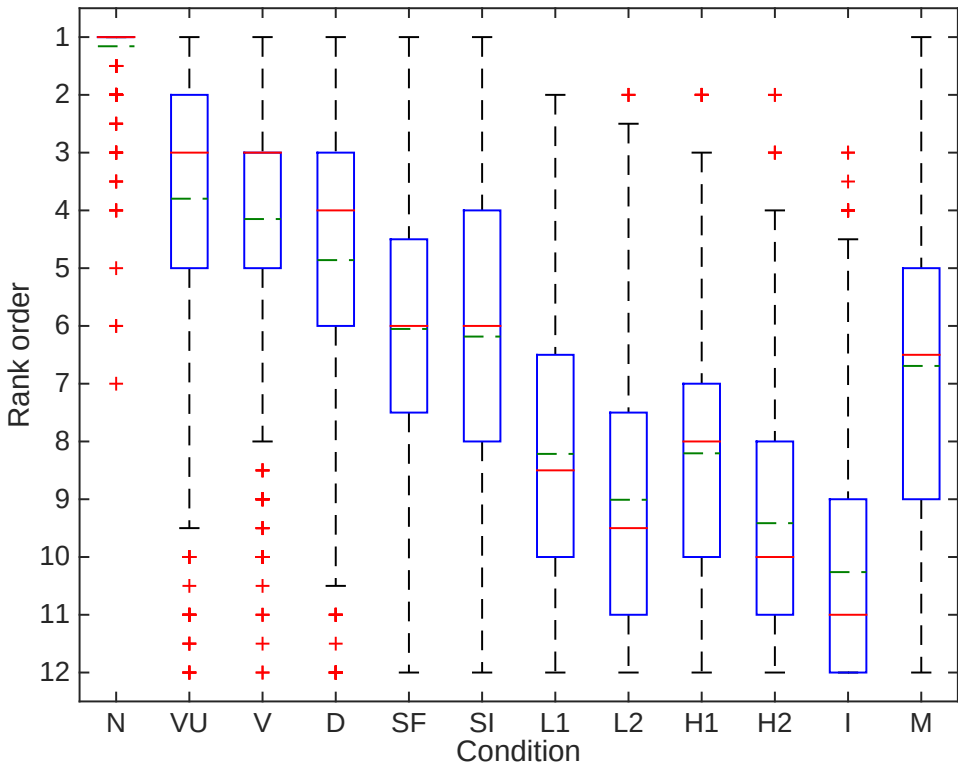


Figure 6.3: *Boxplot of rank order from MUSHRA test. Plot uses the same notation as Figure 4.1.*

naturalness when the diagonal covariance assumption is in force. Secondly when using the mean rather than sampling from the model there is a small drop in naturalness under the source-filter independence assumption. Finally there is a drop in naturalness from using the mean instead of sampling from a model which does not make source-filter independence or diagonal covariance assumptions (condition D).

Within these groupings of conditions, further observations can be drawn. Within the 'vocoded' group of conditions (VU, V and D), the temporal smoothing applied to condition V introduced a slight drop in perceived naturalness. Temporal smoothing was introduced in order to reduce artefacts from vocoded parameters, particularly when different repetitions are substituted to create stimuli under different assumptions (conditions SF, SI, L1, L2, H1, H2, I and M). Although this temporal smoothing has resulted in a drop in ratings when comparing conditions VU and V, it is unclear whether the temporal smoothing has increased the naturalness of subsequent conditions. A further drop in naturalness is observed between conditions V and D, indicating that the DTW alignment, used to time-align the different repetitions of an utterance, introduces slight artefacts. However this slight drop in naturalness is required in order to test

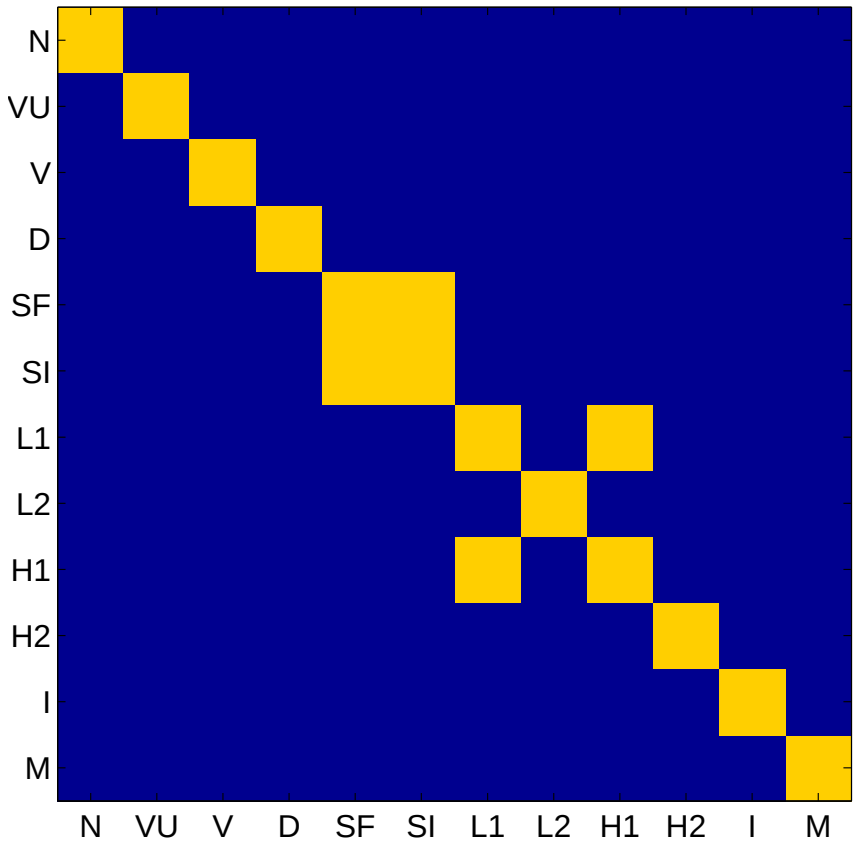


Figure 6.4: *Visualisation of significant differences between systems in terms of rank order using Mann-Whitney U test and the Wilcoxon signed-rank test (p=0.01). Dark blue indicates agreement in significant differences. Yellow indicates agreement in no significant difference.*

the different modelling assumptions subsequent to this condition in the investigation (conditions SF, SI, L1, L2, H1, H2, I and M). The difference in reported naturalness between conditions SF and SI is found to be not significant, this indicates that although a large drop in naturalness is experienced when source and filter parameters are modelled independently (moving from condition D to condition SF), when BAP and f_0 parameters are modelled independently of each other this has no further detrimental effect. A large drop in reported naturalness is observed moving from condition SF to all of the conditions which applied independence assumptions across Mel-cepstra coefficients (i.e., diagonal covariance). Conditions L1, L2, H1, H2 and I, all resulted in a very large drop in naturalness from condition SF when sampling from the model. Within the diagonal covariance group of conditions, it appears that independence assumptions among the lower coefficients (conditions L1 and L2) is more detrimental than independence assumptions among higher coefficients (conditions H1 and H2).

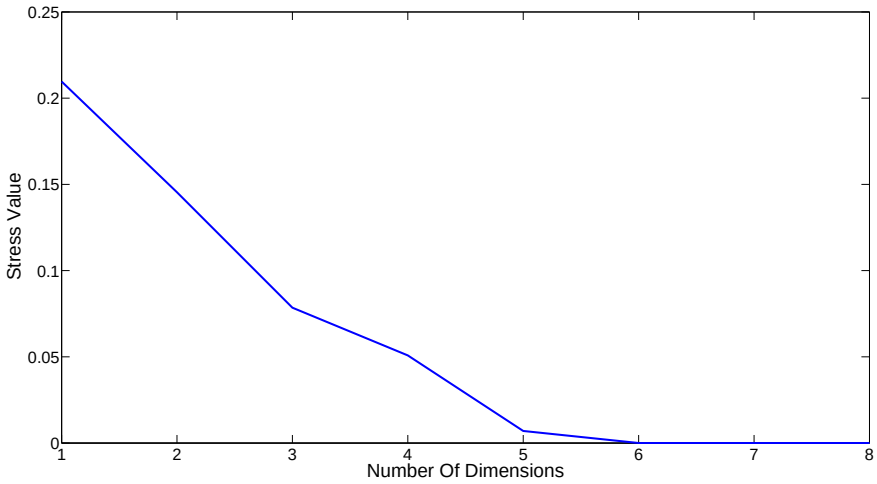


Figure 6.5: *Stress levels return by MDS at different dimensions.*

This is highlighted when comparing conditions L1 and H1. Even though fewer coefficients are assumed to be independent in L1 compared to H1, this reported level of naturalness is very similar (however there is a large detrimental effect observed across all the conditions in this group).

6.6.2 Pairwise listening test

Figure 6.5 shows the stress value when attempting to fit the MDS representation to differing number of dimensions. The stress values across differing number of dimensions do not possess the attributes described in Chapter 1, to warrant the two-dimensional MDS plot to be used for analysis. However, following a comparison of the subsequent plots at two and three dimensions there wasn't a great deal of difference between the points in the plot which is interpretable. Therefore it was decided for simplicity to use the two-dimensional plot. The subsequent MDS plot is shown in Figure 6.6. The responses from this listening test are consistent with the findings of the MUSHRA test. As such the hierarchy of assumptions is reflected in the MDS plot. This hierarchy has been overlaid on top of the MDS plot with arrows, where each arrow represents a further step, in terms of assumptions, away from natural speech. The groupings of conditions seen in the responses from the MUSHRA listening test also appear to be roughly present in the MDS plot.

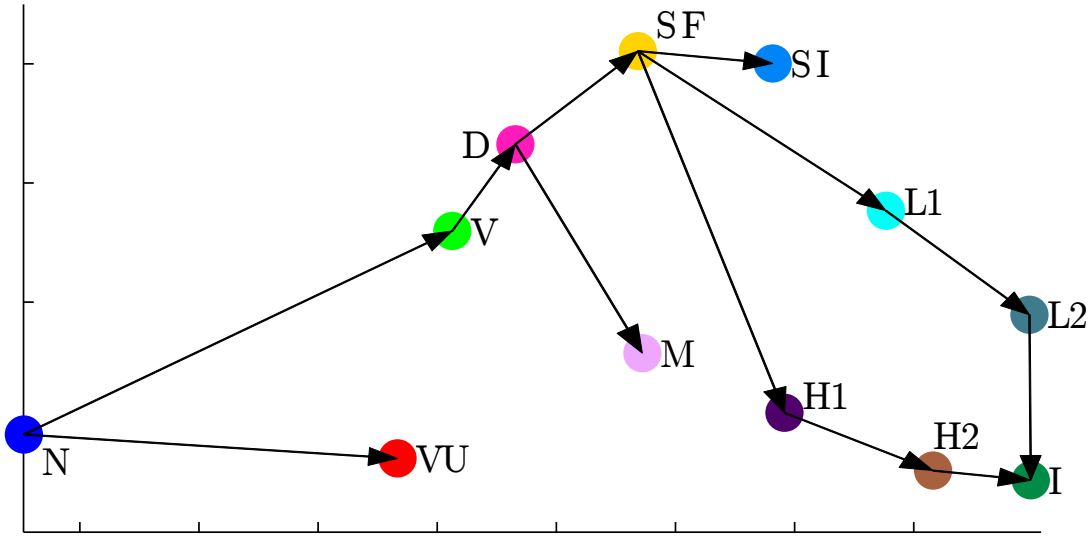


Figure 6.6: *The MDS visualisation of listeners responses at 2 dimensions. Figure appeared in Henter et al. (2014a).*

6.7 Conclusions

This chapter introduced a novel method of investigating the upper-bound performance of SPSS, whilst different common modelling assumptions are in place. Of the assumptions tested, amongst the largest causes of degradations was found to be introduced simply by vocoding the speech signal, indicating that before any modelling has even been performed there is a large drop in naturalness (as was also found to be the case in Chapters 4 & 5). Additional degradations were experienced when 'ideal' sampling from the models is performed; under the assumption that source and filter parameters are independent (as was found to be the case in Chapter 5), and a dramatic drop in reported naturalness was observed when assuming diagonal covariance across any of the filter (Mel-cepstra) coefficients. Additionally when the mean value, rather than sampling, is used from our 'ideal' models (which perform only averaging within matching linguistic contexts at the frame level) we approach the same condition regardless of the independence assumptions put in place in the system (condition M). This condition was found to introduce degradations, but only to a level similar to the source-filter independence assumption; the performance was an improvement on sampling from a model making a diagonal covariance assumption.

The use of a corpus of repeated utterance recordings, such as the REHASP 0.5 corpus presented in this chapter, may be of interest elsewhere in speech synthesis systems. For example, currently SPSS systems are trained in such a way that the one recorded

example of each utterance in the corpus is the example from which objective distance measures of system performance are calculated. The REHASP corpus highlights the many slightly different ways the same text may be read, even though it is in the same neutral speaking style. Therefore it would seem more appropriate for a synthesis system to be deemed a success if it generates speech trajectories which are close to *one* of the many possible realisations of the given utterance, rather than using the distance to a single realisation. This gives a corpus of repeated utterance recordings potential applications in objective scores at training-time and synthesis-time in statistical parametric speech synthesis. The REHASP corpus may also be of interest to increase the understanding of the findings in Chapter 4 on the effect of averaging across examples of matching linguistic contexts. The REHASP corpus allows us to test whether the large perceptual improvement found in Chapter 4 hold when averaging together multiple examples of matching linguistic context or if the large perceptual gain observed came from only averaging together frames from a single example of a unique linguistic context (as was a side-effect of the investigation approach taken in Chapter 4). This investigation is left as future work, however within a standard training corpus of speech the occurrence of multiple examples whose linguistic contexts exactly match is expected to be extremely low.

Chapter 7

Summary of investigations in Part I of the thesis

7.1 Discussion

In Part I of the thesis, a variety of perceptual investigations into possible causes of the reduced quality experienced within statistical parametric speech synthesis (SPSS) systems were conducted. These investigations have provided us with a firmer understanding of what the differing causes of reduced quality within SPSS are, along with the magnitude of each of these contributing factors. Given the findings of the investigations performed in Part I of the thesis, we are better able to apply informed improvements to SPSS, to overcome these issues. This chapter will now summarise these findings and identify which causes will provide the focus of the improvements to be investigated in Part II of the thesis.

7.1.1 Small amounts of temporal smoothing is not harmful

Before work on this thesis began, terms such as 'over-smoothed trajectories' were used in the literature as being a major cause of degraded synthesis quality from SPSS systems. However it was unclear what was meant by this term or whether there is even agreement between different authors. One possible interpretation of what is meant by 'over-smoothed trajectories' is that speech parameter trajectories output from SPSS appear temporally very smooth. To the eye, comparing SPSS-generated trajectories to those of natural vocoded parameters, it is clear which is synthetic and which is natural, due to the synthetic trajectories being much smoother.

These temporally-smooth trajectories are a result of the maximum likelihood parameter generation (MLPG) algorithm, a generation algorithm used in SPSS to link together the different HMM models selected to synthesise the utterance. MLPG looks to optimise the generated trajectory by incorporating delta and delta-delta information about parameter trajectories. The subsequent generated parameter trajectory varies smoothly with time, across the utterance. MLPG reduces the effect of shifts in the generated speech parameters which would otherwise occur at state boundaries, if simply generating from the model mean value.

As many of the claims in the literature pointed towards temporal-smoothness as a contributing factor to reduced quality in SPSS (Zen et al., 2009, Takamichi et al., 2014b), this was an important element to include in the 'continuum of speech' conceptual investigations performed in Part I of the thesis. The trajectories of natural vocoded speech parameters were smoothed using a sliding Hanning window of fixed size, as described in Chapter 3. The effect of temporal smoothing was tested in Chapters 3, 4 & 5. Contrary to claims in the literature, temporal smoothing of the vocal filter speech parameters was found to consistently have little to no effect on the perceived quality of speech. Therefore this hypothesised cause can be removed from further work in this thesis.

The temporal smoothing finding from Part I of the thesis is particularly interesting as the modulation spectrum postfiltering work in literature states that it aims to re-instate temporal detail into temporally over-smoothed trajectories (Takamichi et al., 2014b). Given the findings from Part I of the thesis, it would appear that this method may also be re-instating variance into the generated trajectories, represented by lower frequencies in the modulation spectrum domain. Modulation postfiltering may in fact be making its gains in quality by effectively re-instating the signal variance and not from repairing temporal smoothness. The findings of Part I of the thesis may also tally with the findings of Takamichi et al.. In Takamichi et al. (2015), it was found that, above 50 Hz in modulation spectrum frequency, only noise was being introduced into the generated speech. As a result of this Takamichi et al. placed an upper limit of 50 Hz on the modulation spectrum frequencies which are re-instated. This appears to be consistent with the finding in Part I of the thesis: a large amount of the temporal detail in the vocoded parameters may be noise from the parameter extraction process rather than important detail. By placing this threshold of modulation spectrum restoration, Takamichi et al. may actually be primarily reinstating important elements of variance back into the trajectories rather than temporal detail. Future work of interest with

regards to modulation spectrum research, although outside of the scope of this thesis, may be to explore which modulation spectrum frequencies are important, using the perceptual framework introduced in this thesis.

One issue raised by the finding that temporal smoothing has little to no effect on synthesis quality, is with objective scoring measures. This is because the differences between a natural vocoded trajectory and one which has had temporal smoothing applied to it, appear to be large. However, perceptually, there was found to be little difference. This therefore means objective scores which take a frame-level error measure may give a temporally smoothed trajectory a large error rating whereas listeners rate this speech as being natural.

7.1.2 Generating parameters with correct variance is important

It has been long observed that parameters output from HMM speech synthesis systems have a reduced level of variance than observed in natural vocoded speech. This is due to averaging across many examples (frames) within clusters in the decision tree of the standard HMM synthesis system. This averaging results in convergence towards the mean of generated speech parameters, removing occurrences of large deviations from this average value which are present within natural speech. Hence, improvements in naturalness are observed from the use of global variance (GV) as a post-modelling tool such that the use of GV within HMM speech synthesis systems has become standard (Zen et al., 2007a, 2009, King, 2011).

However attempting to match a variance level, which is predicted to be the correct amount across the utterance to be generated, provides no guarantees as to how close the subsequent parameter trajectories would match a natural example of the utterance. Therefore the investigations undertaken in Part I observed the effect of incorrect variance of speech parameters as a result of overestimated or underestimated GV. This simulation effect was implemented using variance scaling of utterance-level speech parameter trajectories, as described in Chapter 3. The effect of utterance-level parameter variance was investigated in Chapters 3, 4 & 5. As expected, given reported gains in the literature (Toda and Tokuda, 2007, Zen et al., 2009), generating speech parameters with the correct variance level was found to have a large effect on the quality of synthesised speech. The investigations in this thesis extended this knowledge by also observing that erring on the side of too much variance appears to be preferred to underestimated parameter variance. This added knowledge may mean that when performing

GV, we can afford to slightly over-compensate the variance to restore in the speech signal; however further investigation as to whether this helps in SPSS is needed.

7.1.3 Vocoding alone introduces a noticeable drop in quality

Vocoding is typically used to extract parameters of speech. These are then converted into parameters which are more suitable for modelling within SPSS systems. Therefore the accuracy of parameter extraction, along with how invertible the vocoding step is, places a large constraint on the quality of speech achievable. Given that these speech parameters are then used to train the models used for synthesis, it appears logical to surmise that the quality of the vocoder acts as an upper-bound to the quality achievable from SPSS. As a result of this, natural speech along with natural vocoded speech (copy-synthesis) were included as conditions within a number of the perceptual tests conducted in Chapters 4, 5 & 6 of Part I of the thesis. These different tests observed the effect of vocoding across a range of vocoders widely used for SPSS: STRAIGHT (Kawahara, 2006), GlottHMM (Raitio et al., 2011b) and the PSFT vocoder. All of these vocoders were found to introduce a perceptually noticeable drop in quality before any modelling has even taken place. The use of this range of different vocoders is partly a consequence of collaborations with different researchers, however findings across these different vocoders are consistent. The following effects were tested using multiple vocoders:

- Temporal smoothing and global variance were tested in Chapters 3, 4 & 5 using STRAIGHT, PSFT & GlottHMM vocoders.
- The effect of independently modelling of parameter streams was tested in Chapters 5 & 6 using GlottHMM & STRAIGHT vocoders.
- The effect of vocoding was tested in Chapters 4, 5 & 6 using the PSFT, GlottHMM & STRAIGHT vocoders.

This adds confidence to the conclusions drawn in Part I of the thesis, removing concerns about vocoder-specific findings.

One interpretation of the finding that vocoding introduces a perceptually noticeable drop in quality is that it seems reasonable to assume that future improvements in vocoders will in turn lead to better quality of SPSS. This is provided that they allow for extraction of speech parameters suitable for modelling. This observation is made given that SPSS performance appears to be tethered to that of the vocoder used.

An alternative consequence of these findings would be to explore synthesis methods which aim to overcome the issue of reduced quality following vocoding, by removing the current model-friendly vocoders. These include investigating the use of sinusoidal vocoders within SPSS, as in Stylianou (1996), Erro et al. (2007) and Hu et al. (2014b), or investigating the domain of unit selection where no vocoding takes place.

7.1.4 Findings consistent across different parametrisations tested

As mentioned in Section 7.1.3, vocoding is performed in order to extract components of speech. However the parameters, representing the components of speech, returned by the vocoder are unsuitable for modelling directly. Therefore the parameters output by the vocoder are then transformed into speech parametrisations which are more suitable for modelling (i.e. the data is reduced in dimensionality and decorrelated such that a statistical model is able to model the parameters with a higher degree of accuracy). It was of interest to identify how parameter-specific the perceptual findings from the investigations in Part I of the thesis are, as well as seeing if one parametrisation performs better than another. As such, the experiments in Chapter 4 were run on the two different popular parametrisations predominantly used in speech synthesis literature: Mel-cepstra and Mel-LSP. The findings of these listening tests indicated that similar perceptual effects occur with both parametrisations of speech, across the range conditions tested. This indicates that the perceptual findings are not as a direct result of the parametrisation in use but of more fundamental elements of the speech synthesis system, namely the vocoder and the modelling of the speech parameters.

7.1.5 Independent modelling of parameter streams introduces large drop in quality

In a standard HMM synthesis system, the different speech parameter streams, output by the vocoder and converted to a parametrisation suitable for modelling, are usually clustered using separate state-dependant decision trees. This results in independent clustering of linguistic contexts for each of these different streams, which may result in different linguistic contexts being clustered together. In order to test the effect of this design approach, perceptual testing was performed in Chapters 5 & 6. Chapter 5 investigated the contribution to the quality of overall synthesis quality from modelling the source and filter parameters. Using the GlottHMM vocoder, this investigation was able to alter source and filter independently and monitor where losses in quality were expe-

rienced. Chapter 6 investigated the upper-bound performance of SPSS when making the source-filter independence assumption, using a corpus of repeated natural speech (REHASP 0.5). In the perceptual testing conducted in Chapter 6, listener responses showed a drop in naturalness rating where the assumption of independent parameter stream modelling was present. One interpretation of these findings are that independent parameter stream modelling results in a lack of consistency (i.e., covariance) between different speech parameter streams at synthesis time, resulting in a drop in naturalness. The lack of covariance modelling is not only present as a result of modelling speech parameters with separate decision trees, it is also present in the SPSS parameter generation. MLPG does not use covariance between different parameter streams when generating parameter trajectories, resulting in further reduced dependency between generated parameter streams.

7.1.6 Diagonal covariance modelling introduces large perceptual drop in quality

In a standard HMM synthesis system, the different coefficients within a parameter stream (e.g. Mel-cepstral features) are often modelled using diagonal covariance. This means that the relationship between coefficients is not modelled, instead each coefficient is modelled independently. To test what effect this had on SPSS performance, perceptual testing was performed in Chapter 6. This investigation observed the upper-bound naturalness achievable when making the diagonal covariance assumption, using a corpus of repeated natural speech (REHASP 0.5). Perceptual testing indicated that this lack of covariance modelling between parameter coefficients does indeed lower the naturalness of synthesised speech when sampling from the model, however this is not the case when using the mean. Following this observation, it would be of future interest to investigate how well full covariance modelling within SPSS is able to model these dependencies relative to the upper-bound investigated in Chapter 6 when sampling from the model (the distance between conditions SF and I).

7.1.7 Averaging within matching linguistic contexts is much less harmful than across differing linguistic contexts

In standard HMM synthesis, decision tree regression is used to cluster the linguistic contexts from the training data in order to account for inevitable unseen linguistic con-

texts at synthesis-time. At training-time a criterion (typically maximum likelihood) is used to determine which questions about the linguistic context should be used to split the speech data within the decision tree. There is a stopping condition (typically based on minimum description length) associated with the decision tree to stop the clusters of linguistic contexts from being excessively split. Excessive splitting of linguistic contexts within the decision tree would result in over-fitting to the training data. Following this stopping condition being met, a HMM state is created at each leaf node in the decision tree. The stopping condition ensures that the models are able to account for unseen linguistic contexts by generalising across the seen linguistic contexts.

As a result of the decision tree regression, differing linguistic contexts are modelled together. These linguistic contexts are deemed by the stopping condition to be close to each other such that modelling them together will generalise effectively for unseen contexts. However as these linguistic contexts are in fact slightly different from each other, in terms of not having exactly matching linguistic context strings, this can result in distorted models of speech being created. In order to measure the effect of this a 'pseudo-HMM' condition was introduced in the investigation in Chapter 4 of the thesis. The 'pseudo-HMM' condition used standard HMMs for synthesis with the exception of using an 'ideal' model mean value which was calculated from frames whose linguistic contexts exactly matched. Comparing the perceptual responses from listeners between the 'pseudo-HMM' system and standard HMM synthesis, it is apparent that averaging across frames with matching linguistic contexts results in synthesised speech which is very natural (perceptually very close to natural vocoded speech). Whereas in standard HMM synthesis where differing linguistic contexts are averaged together, synthesis performance appears to suffer from distortions in the models created due to the presence of slightly differing linguistic contexts. Following these findings it appears that, by performing better averaging in the statistical models to be used for synthesis, large gains can be made.

7.2 Concluding remarks

Part I of the thesis has focused on the contributing factors to reduced synthesis quality within HMM speech synthesis. This investigation has deliberately aimed to exclude prosodic elements, as prosody is a large field of research in its own right. To include investigations in this field of research would require a huge additional amount of perceptual testing in order to reach firm conclusions. However it is possible that

researchers in the field of prosody may wish to re-run the methodology presented in Part I of this thesis in the future, focusing instead on prosodic effects.

Given the findings from Part I of the thesis, there appears to be clear indication that addressing two of the found causes should be able to mitigate some of the main shortcomings of SPSS:

1. Remove averaging across differing linguistic contexts and instead only perform averaging across matching linguistic contexts. This will be investigated in Chapter 9.

2. Remove the effect of vocoding. This will be investigated in Chapter 10.

Addressing these issues will therefore be the main aim in Part II of the thesis.

Part II

Motivated improvements to HMM synthesis

Chapter 8

Updated background

8.1 Advances in speech synthesis

At this point of the thesis, the use of neural networks for speech synthesis had re-emerged. Deep neural network (DNN) regression of speech parameters was widely reported in the literature to lead to increased naturalness within the statistical parametric speech synthesis (SPSS) paradigm (Zen et al., 2013, Zen, 2015). This lead to DNN regression-based synthesis systems becoming more prevalent than the decision tree regression-based HMM synthesis systems tested in Part I of the thesis. This chapter will provide an overview of DNN methods. Feed-forward DNNs are used in Part II of the thesis, so only these will be discussed in depth here.

8.1.1 Feed-forward deep neural network (DNN)

The feed-forward DNN configuration discussed in this chapter is the system described in Wu et al. (2015). This is the DNN configuration which will be used in Chapters 9 & 10. As with the standard HMM synthesis architecture discussed in Chapter 1, the outputs of the DNN are speech parameters (Mel-cepstra, log-f_0 and BAPs) with the addition of a binary feature denoting whether the current frame is voiced or unvoiced. All speech parameters are predicted at the same time in this feed-forward DNN architecture. The inputs to the DNN are 601 linguistic context features. 592 of which are binary, derived from a subset of the questions about linguistic contexts used for decision tree clustering in the HMM synthesis system described in Chapter 1. There are also 9 numerical features, as described in Watts et al. (2016a), these are: the frame position within the current state (as a fraction counting from the beginning

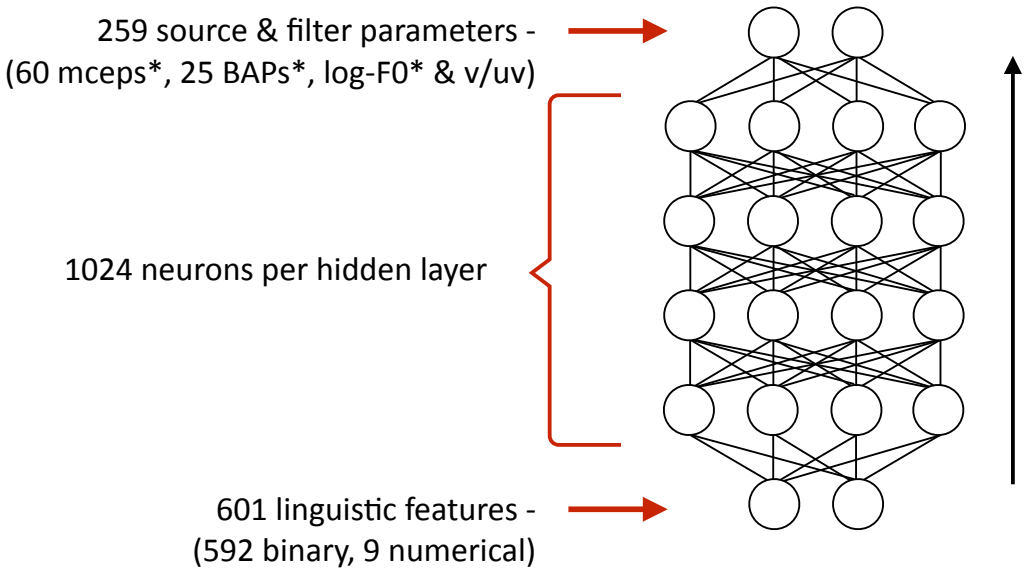


Figure 8.1: *Illustration of feed-forward deep neural network. "*" denotes that static, delta and delta-delta attributes are output.*

and from the end of the state) and within the current phoneme (as a fraction counting from the beginning and from the end of the phoneme), state position within the current phoneme (counting from the beginning of the phoneme and counting backwards from the phoneme), state duration, phoneme duration and fraction of the current phoneme made up by the current state.

Figure 8.1 shows an overview of the feed-forward DNN architecture. At each layer of the DNN there are several neurons (determined by how 'wide' the current layer is) which each compute an activation function. The hyperbolic tangent activation function is the non-linear activation function used in the DNN in subsequent chapters in this thesis, however other activation functions are available. Each of the neurons in the previous layer is fully connected to each of the neurons in the current layer. Given a connection weight for the value output from the activation function from each of the neurons in the previous layer to the current neuron, the multiplications of the activations from the neurons in the previous layer with their associated weights are summed and a neuron bias term is added to form the input to the current neuron's activation function. The activation function of each of the neurons is a non-linear

compressive, differentiable transform of the pre-activation value. The combination of these various non-linear transforms in a DNN results in the construction of a very complex function. It is this potential which makes DNNs very attractive in order to provide a complex mapping between the input (linguistic context) and output (speech parameters). The layer before the output of the DNN is a linear layer. This layer provides our speech parameters at the output of our DNN. The output of the DNN is therefore the result of several layers of many neurons performing non-linear transforms from the input.

The problem remains as how to best train the DNN in order for it to learn the relationship between linguistic contexts and speech parameters. The initialisation of the DNN is very important as to how the system will perform. In this thesis the weights at each node were initialised to random values from a normal distribution with a mean of zero and a standard deviation which is conversely related to the number of nodes which feed into the current node. The biases are initialised at zero. The input features were normalised to the range between 0.01 and 0.99, while the output features were normalised to have zero mean and unit variance ($\sigma^2 = 1$) (Wu et al., 2015). For training a number of frames from the training set are passed through the DNN. These frames are random frames from the training data. The output from the DNN is compared with the real examples from the training data. The error between these is computed as a mean squared error function (Ling et al., 2015). The error is back-propagated through the DNN, using stochastic gradient descent (Duda et al., 2001). This is done in order for the weights and biases across the DNN to be updated. Stochastic gradient descent, updates the DNN on part of the training data. This is because it is computationally cheaper as updates to the DNN are performed many times per single pass through the entire training set (epoch). By updating the DNN many times per epoch, less epochs are required to train the DNN. For stochastic gradient descent, the gradients of the error with respect to the weights and biases are computed. Following the calculation of the gradients, the weights and biases are updated according to the learning rate. A validation set of utterances, unseen in the training and testing utterances, is used to measure how well the DNN is generalising for unseen utterances during training. The error across the utterances in the validation set is calculated. The calculated error across the validation set is not used to update the weights or bias values in DNN, but instead to judge when training can be stopped.

For the DNN used in the remaining chapters in this thesis, the parameter estimations are made at the frame-level, rather than the state-level predictions used in the

standard HMM synthesis system discussed in Chapter 1. The potential influence of such differences in standard system configurations for DNN and HMM synthesis systems will be discussed further in Chapter 11. The frame-wise parameter predictions from the DNN include static, delta and delta-delta attributes. Generation of the parameter trajectories is done using MLPG, as described in Chapter 1, by considering the output of the DNN to be the mean vector of a frame-level Gaussian model. The covariance matrix used for MLPG is calculated using all training data. The generated parameter trajectories are passed back through the vocoder to provide the time-domain waveform.

8.1.2 Other DNN architectures

There are many alternative DNN architectures which can be used for speech synthesis: for example, recurrent neural networks (RNNs) are becoming more common (Zen, 2015). In an RNN, in addition to each of the neurons at each layer being fully connected to the neurons at the previous layer, like in the feed-forward DNN, there are also connections to activations from neurons at the previous frame. This introduces dependencies from previous frames meaning that the DNN is no longer generating parameter values at each frame independent of the values of surrounding frames. The long short-term memory (LSTM) architecture is the most commonly used RNN architecture (Zen and Sak, 2015). Bi-directional LSTMs extend this frame-level dependency further by having connections in neurons between following frames as well as previous frames (Fan et al., 2014), however bi-directional LSTMs are much more computationally expensive.

8.2 Summary

Although DNNs have been found to improve the performance of SPSS, many of the assumptions investigated in Part I of the thesis are still in place. Part II of the thesis will now look at using the findings of the investigation in Part I of the thesis to make improvements to the quality of synthesis, for the case of HMMs, but it is believed that some of the findings also apply to DNNs.

Chapter 9

Avoid performing averaging across differing linguistic contexts - rich-context synthesis

This chapter is an expanded version of the work in Merritt et al. (2015b) and therefore the text is closely related to that.

This work was completed in collaboration with others. Discussion of ideas was done between myself, Junichi Yamagishi, Zhizheng Wu, Oliver Watts and Simon King. The code for the feed-forward DNN system used was provided by Zhizheng Wu. The code for running and analysing the MUSHRA test was provided by Gustav Eje Henter. Adaptations of the HTS (Zen et al., 2007a) scripts to produce the rich-context synthesis was done by myself. The methods for selecting rich-context models was implemented by myself.

9.1 Motivation

Following the perceptual tests performed in Part I of the thesis, the following were found to be significantly contributing to the reduced quality of HMM speech synthesis:

- Generating parameter trajectories with incorrect variance.
- Parametrisation of speech (vocoding).
- Independent modelling of parameter streams.

- Diagonal covariance (i.e., no covariance modelling) between spectral parameter coefficients.

- Averaging across differing linguistic contexts was found to be perceptually much more harmful than only averaging within matching linguistic contexts.

Of these findings, the gains in quality observed in Chapter 4 are of particular interest. Chapter 4 investigated the effects of within matching linguistic context averaging with an idealised 'pseudo-HMM' condition. Standard HMM speech synthesis systems perform averaging across differing contexts as part of the decision tree regression performed. The decision tree regression within HMM synthesis is designed to generalise across seen linguistic contexts in order to account for the inevitable linguistic contexts which are unseen in the training data. However by comparing listener responses of the pseudo-HMM condition to those of the standard HMM speech synthesis system, it is clear that averaging across differing linguistic contexts degrades speech quality. The perceptual findings in Chapter 4 indicate that by implementing a system which performs averaging only across matching linguistic contexts, the speech synthesis quality produced is greatly improved. It is this finding which primarily motivates the investigation undertaken in this chapter, that of attempting to improve the quality of HMM synthesis by removing averaging across differing linguistic contexts.

In Chapter 4, a large perceptual degradation was reported when moving from the idealised pseudo-HMM condition to a standard decision tree-clustered HMM synthesis system (built as per the HTS demo recipe). In the pseudo-HMM condition, the mean parameter values were calculated only across examples which had exactly the same linguistic context, whereas the standard HMM synthesis system averaged across the examples within each of the leaf nodes in the decision tree. The pseudo-HMM system calculated the Gaussian mean values (for static, delta and delta-delta acoustic features) from the acoustic features (i.e., vocoder parameters) of a recording of the test sentence. The synthetic speech was then created by using MLPG on this sequence of Gaussian means, with variance values coming from the conventional HMMs. Such a system results in a parametric system which performs all the standard steps of the HMM synthesis system with the exception of the use of this 'ideal' model mean value. Of course, this oracle system is of no use for actual text-to-speech because it is impossible to have examples in the training data that exactly match all required contexts for every test sentence. However, the findings motivate the system presented in this chapter that (like existing so-called "rich-context" systems (Yan et al., 2009)) avoids

averaging across differing linguistic contexts to train the Gaussian means.

9.2 Previous work

There are two notable examples of systems that also aim to remove the effects of averaging across differing linguistic contexts. The first is simply unit selection synthesis, where individual tokens are used, without averaging. Unit selection synthesis systems also avoid vocoding, another limitation on the quality of parametric speech synthesis as identified in Part I of the thesis; this will be investigated in Chapter 10. The other example is rich-context statistical parametric speech synthesis (Yan et al., 2009, 2010, Takamichi et al., 2012, 2014c).

The term 'rich-context' refers to models which are trained only on samples where the linguistic contexts match exactly and therefore avoids averaging across differing contexts. The primary example is Yan et al. (2009), in which Gaussian mean values are calculated within each unique context found in the training data, with variance values being tied in the usual way. Rich-context synthesis systems would therefore appear to be very close to the previously investigated pseudo-HMM condition in Chapter 4. In practice, such a system is easy to derive from a conventional tied system, simply by untying all parameters, then performing further training in which only the means are updated. The problem with rich-context models is how to select suitable models to use at synthesis time, given that exact matches to the required contexts within a test sentence are extremely unlikely to be available. Rich-context models are sparsely trained, meaning they no longer generalise over seen linguistic contexts in the training data to predict features for unseen linguistic contexts, as in standard HMM synthesis. Instead, the task now for generating unseen linguistic contexts is to identify the *closest* linguistic context which was present in the training data and use the corresponding rich-context model for synthesis.

9.3 Conventional rich-context system

The system introduced in Yan et al. (2009) uses the distribution (i.e., Gaussian) selected by the standard tied decision tree as a reference. It then finds the closest untied rich-context model (from a pre-selected subset of all possible models) to this reference, using the upper-bound of inequation 9.2 to compute divergence between the reference distribution and each of the rich context models. This equation, as described in Liang

et al. (2008), is an adapted version of Kullback-Leibler divergence (KLD) for calculating divergence between multi-space probability distribution HMMs (MSD-HMMs) and can therefore be applied to both spectrum (S) and pitch ($f0$) parameters independently.

$$D_{KL}^{S+f0}(p||q) = D_{KL}^{S}(p||q) + D_{KL}^{f0}(p||q), \tag{9.1}$$

where

$$\begin{aligned} D_{KL}(p||q) \leq (w_0^p - w_0^q)\log\frac{w_0^p}{w_0^q} + (w_1^p - w_1^q)\log\frac{w_1^p}{w_1^q} \\ +\frac{1}{2}tr\{(w_1^p\Sigma_p^{-1} + w_1^q\Sigma_q^{-1})(\mu_p - \mu_q)(\mu_p - \mu_q)^\top \\ +w_1^p(\Sigma_p\Sigma_q^{-1} - I) + w_1^q(\Sigma_q\Sigma_p^{-1} - I)\} \\ +\frac{1}{2}(w_1^q - w_1^p)\log|\,\Sigma_p\Sigma_q^{-1}\,|, \end{aligned} \tag{9.2}$$

p and q are the reference and pre-selected HMM states respectively, w_0 and w_1 are the prior probabilities of unvoiced and voiced respectively (for spectrum $w_0 \equiv 0$ and $w_1 \equiv 1$), μ and Σ are the mean and covariance of the Gaussian distributions respectively and $|\cdot|$ indicates the determinant of a matrix. The divergences for spectrum ($D_{KL}^{S}(p||q)$) and pitch parameters ($D_{KL}^{f0}(p||q)$) are then summed together using equation 9.1 to provide the final divergence score $D_{KL}^{S+f0}(p||q)$. These equations are applied in a state-wise fashion. All divergence values across the test phoneme (5 states) are added together to arrive at a single value per phoneme. The rich-context model (i.e., all 5 states in that model come from the same context) with lowest total divergence is then selected.

9.3.1 Implementation issues

One issue encountered when using this formula with rich context models, but unreported in Yan et al. (2009), is that rich context models are generally either completely voiced or unvoiced, making either w_0 or w_1 equal to zero. In our replication of Yan et al. (2009), where this occurs, a small number (0.001) was added to or subtracted from w_0 and w_1 to ensure a division by zero never takes place. The problem of zero divisions also appears in the spectrum calculation where $w_0 \equiv 0$ and $w_1 \equiv 1$; in this case $(w_0^p - w_0^q)log\frac{w_0^p}{w_0^q}$ was set to 0.

Also, the adaptation to enable KLD to be used for MSD-HMMs means that the divergence measure is no longer symmetric; so in our work, $D_{KL}(p||q)$ and $D_{KL}(q||p)$

were averaged together to give the final divergence score. This was not mentioned in Yan et al. (2009), so it can only be assumed that this was how the original implementation was done.

9.3.2 Critique

The reference distribution used in Yan et al. (2009) is a standard tied model. That is, the system chooses the rich-context model that is most similar to the model that would be used in a conventional system. This is counter-intuitive. As we know from the perceptual investigations undertaken in Part I of the thesis, this tied model is known to be of poor quality as a result of averaging across differing linguistic contexts. Therefore the tied model would seem to be a poor reference for rich-context model selection. The whole point of using rich-context models is to get away from the tied model, not to find a model that is as close as possible to it.

As mentioned above, the system in Yan et al. (2009) selected only from a subset of all possible rich-context models: only contexts matching the triphone of the target linguistic context, and if no matches are available this is expanded to biphone match. The need for pre-selection was given as leading to a 'reasonable size of the search space' (Yan et al., 2009, page 1757).

9.4 Proposed bottleneck-driven system

The proposed system is inspired by that in Yan et al. (2009), however it does not use the tied model as a reference for rich-context model selection. Instead, it performs selection using an acoustically-supervised embedding of the linguistic context, which we derive from the bottleneck layer of a Deep Neural Network (DNN) speech synthesis model (Wu et al., 2015)[1].

The activations at the bottleneck layer of this network comprise a very compact (e.g., 32-dimensional) feature vector that has been learnt over the training data; such a feature vector is often termed an 'embedding' (Bengio and Heigold, 2014). As such, from this point onwards the activations at the bottleneck layer will be referred to as context embeddings.

Each unique input to the DNN (i.e., each unique linguistic context) leads to a particular context embedding. That is, we can derive a compact context embedding rep-

[1]The code for implementing this system was kindly provided by Zhizheng Wu.

resentation of any linguistic context, including those not seen in the training data. We use distance in this context embedding space as the way to select rich-context models at synthesis time. The DNN-derived context embedding is essentially a compression of the linguistic features, but importantly one that has been learned in conjunction with predicting the acoustics. So, for example, acoustically-irrelevant linguistic features will be ignored, and other features will be 'de-noised' and de-correlated. Because part of the linguistic feature set provided as input to the DNN includes a frame count within the current state, the context embeddings have frame-varying values. This allows for the rich-context model search to account for distributions of context embedding values.

Using the speech parameter space to calculate perceptual distance can be referred to as acoustic space formulation (ASF) (Taylor, 2009). This is effectively what the system in Yan et al. (2009) does. Taylor has previously discussed this, however mentioned that they were 'uneasy about the use of cepstral space to represent the perceptual space' (Taylor, 2006a, page 2041). In the proposed system, an embedding of the linguistic space, as learnt by a DNN, is used instead.

The proposed context embedding-guided rich-context system has the added benefit that we are no longer constrained by needing to use speech parameters at the output layer of the DNN, since it is to be used only for deriving the bottleneck features. For example, we could use perceptually-motivated features instead of vocoder parameters; this is future work.

Various measures could be used, at synthesis time, to find the closest rich-context model (in context embedding space) for an unseen context, this can be done in a number of ways. Here I present two possibilities: Euclidean distance and KLD. For both of the linguistic context distance measures used, the selection is of a single model (i.e., each of the state-wise models for the different speech parameter streams comes from the same rich-context). This decision was by design as it is believed that this consistency between the different parameter streams, in terms of the matching frames from the training data used to train the rich-context models, results in improved synthesis quality. Selection of rich-context models is also possible at the state-level of course, however it was not included in this investigation as it was believed that stability across the phone was most likely to yield improved naturalness. The investigation of state-wise rich-context model selection is left as future work. First, I give more details of how the bottleneck features (context embeddings) are derived.

9.4.1 Bottleneck features

To generate bottleneck features, we used a feed-forward neural network with six hidden layers. Each layer had 1024 hidden units except that the second hidden layer (the second layer being closer to the linguistic features at the input of the network, rather than the speech parameters at the output) was set as a bottleneck layer which had only 32 hidden units. This was because in preliminary experiments by Wu et al. it was found that using the second layer as the bottleneck layer achieved the best performance for DNN-based speech synthesis in terms of acoustic feature distortions. More details about the input and output features and implementations of the DNN can be found in Wu et al. (2015). The input to the DNN includes HMM state-position (i.e., sub-phoneme) and frame-within-current-state counter-based features. Bottleneck features were pre-computed for all frames in the training data using a forward pass, and the mean and variance of the features was computed per rich-context HMM state; these distributions were then stored with the rich-context HMM.

9.4.2 Visualising and interpreting context embeddings

Before using the context embeddings for selecting rich-context models, an initial analysis of the properties of the context embeddings was made. For this, the average value of the 32-dimensional embedding features was calculated for each of the centre-phone identity examples across the training data. The Euclidean distances between each of these average points were then mapped into a visual space using multidimensional scaling (MDS). The stress levels when applying MDS at differing numbers of dimensions are shown in Figure 9.1. Based on the principles for selecting an appropriate number of dimensions for visualising data, as described in Chapter 1, three dimensional MDS was selected for visualisation; these mappings are shown in Figures 9.2 & 9.3. It must be stressed that these visualisations are only for inspection out of interest and sanity checking their suitability for the task of selecting rich-context models.

These MDS mappings show that the context embedding features appear to be learning reasonable characteristics of where these phonemes sit relative to each other. As such, phonemes which are similar in place or manner of articulation are reflected sensibly in their placement in this MDS space. For example the embedding space seems to make clear distinctions between vowels and consonants and between voiced and unvoiced sounds. The context embedding features therefore appear suitable for performing the task of searching for closest seen linguistic contexts from the training data. This

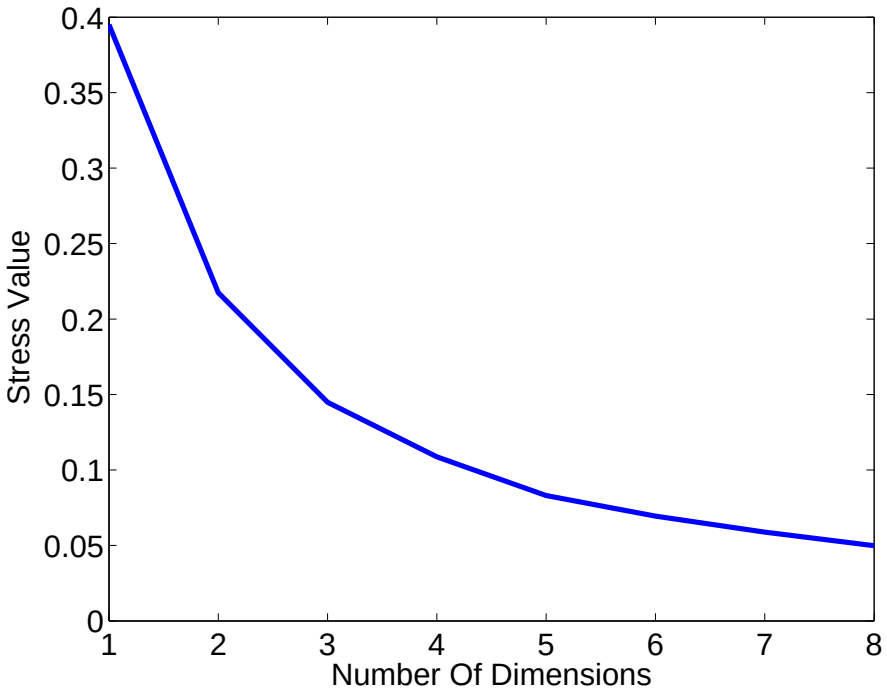


Figure 9.1: *Stress levels returned by MDS when attempting to fit the Euclidean distances between the average embedding value of each of the centre-phone identities to different numbers of dimensions.*

search in the context embedding space will lead to the use of the rich-context model associated with the selected linguistic context.

The visualisations in this section have shown us that the context embedding features appear to distinguish the distance between different linguistic contexts sensibly. These average phoneme values and the resulting MDS space will not be used further. From this point on the frame-level 32-dimensional features as output by the feed-forward DNN will be used.

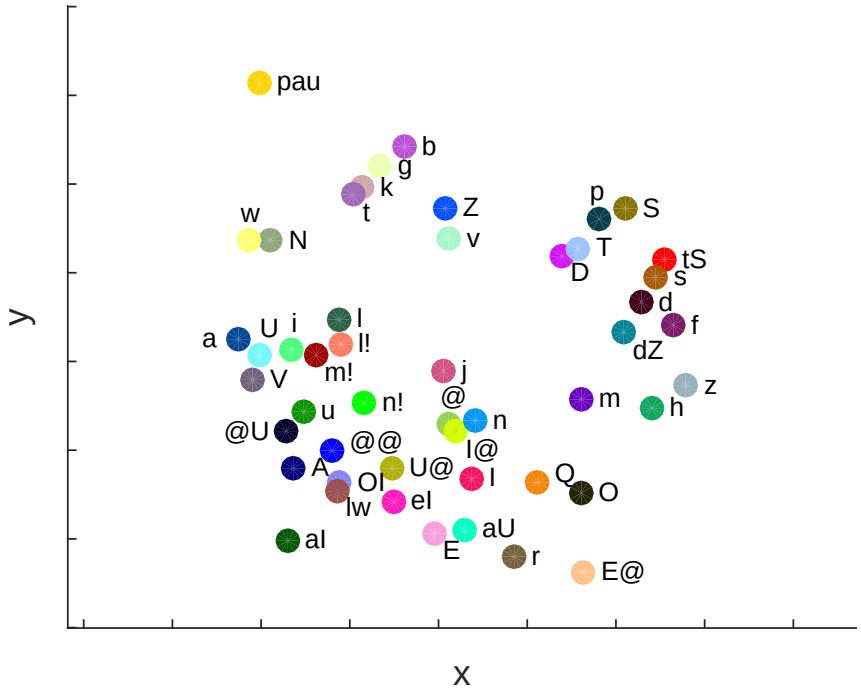


Figure 9.2: *MDS of the distance between average embedding representation per centre phone identity is performed at 3 dimensions. Here the x,y projection is shown.*

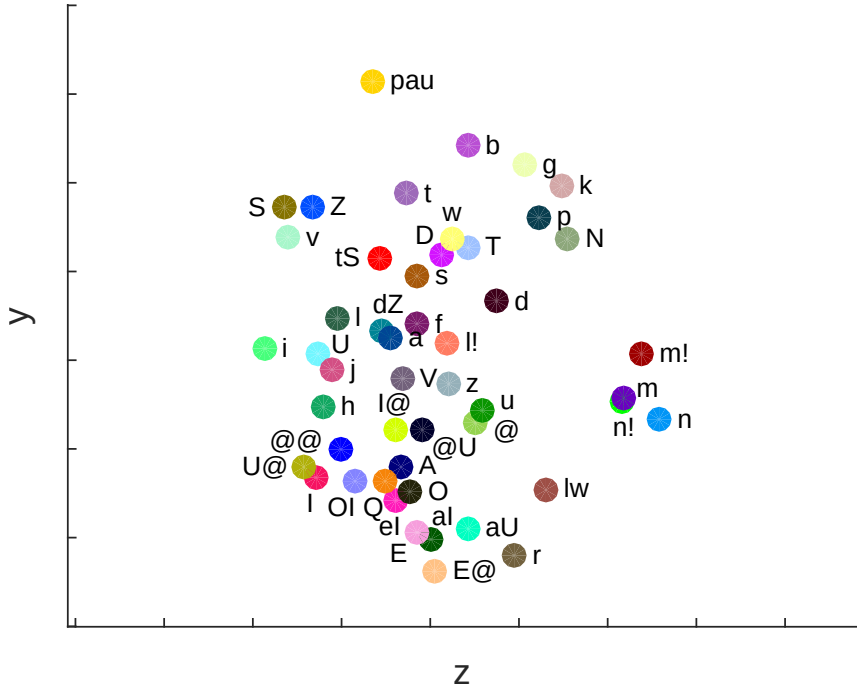


Figure 9.3: *MDS of the distance between average embedding representation per centre phone identity is performed at 3 dimensions. Here the z,y projection is shown.*

9.4.3 Euclidean distance selection

The Euclidean distance between the context embeddings computed for a given state in the test sentence, and the stored context embeddings for each of the rich-context states is:

$$D(b,g) = \sum_{n=1}^{N} || \, b_n - \mu_g \, ||_2, \tag{9.3}$$

where b is the frame-level sequence of context embeddings for the current state in the test sentence, g is the Gaussian distribution (with mean μ_g) of context embeddings for a candidate rich-context model and N is the duration (in frames) of the current test sentence HMM state. The distance for each state in the current phoneme is summed and the phone-sized rich-context model with the smallest Euclidean distance is selected (i.e., all 5 states are taken from the same rich-context model).

9.4.4 Kullback-Leibler divergence selection

The KLD (Hershey and Olsen, 2007) between distribution b of the context embeddings computed over the frames corresponding to a given state in the test sentence, and the stored context embedding distribution g, is calculated as:

$$D_{KL}(b||g) = \frac{1}{2}[log\frac{|\,\Sigma_g\,|}{|\,\Sigma_b\,|} + Tr[\Sigma_g^{-1}\Sigma_b] - d$$

$$+(\mu_b - \mu_g)^T \Sigma_g^{-1}(\mu_b - \mu_g)], \tag{9.4}$$

where μ and Σ are mean and covariance and d is the dimensionality (32 in this case) of the context embeddings. As with the Euclidean distance, the KLD for each sub-phonetic state is summed over the phoneme and the closest model chosen. Variance values were floored to 1% of the global variance. A symmetric version of KLD was used in practice: the average of $D_{KL}(b||g)$ and $D_{KL}(g||b)$.

Table 9.1: *Frequency count of the 60262 unique contexts in the training data*

Frequency	1	2	3
# of rich-contexts	60166	94	2

9.4.5 Rich-context occupancy

The number of occurrences of each of the 60262 unique contexts present in the training data are shown in Table 9.1. These give an idea as to the occupancies of the rich-context models used for this investigation[2]. The counts in Table 9.1 show that the rich-context models use very few instances to compute model mean values from.

9.5 Experiments

9.5.1 Implementation

A variety of system configurations were built, shown in Table 9.2, and compared in a listening test. We created a best-effort replication of the system described in Yan et al. (2009), one with tri-phone (backing off to bi-phone where necessary) pre-selection as per the original system and another with more relaxed bi-phone (backing off to mono-phone where necessary) pre-selection. The latter system has a wider set of rich-context models to select from, per test sentence phoneme, and so should be able to choose a model that is closer to the tied model reference. For that reason, we hypothesise that this will actually sound worse than the more constrained system (even though the reasons for pre-selection given in Yan et al. (2009) were only in regard of computational cost). The average pre-selection candidate list size over the test sentences is shown in Table 9.3. The general system configurations across all systems are shown in Table 9.4. For rich-context model selection, where an exact match exists between the target linguistic context and the rich-context models from the training data, this rich-context model was selected. Otherwise the rich-context model selection methods described in this chapter were used to select the closest rich-context model that was seen in the training data. Note that due to the large number of elements within the linguistic context, an exact match is extremely unlikely.

No pre-selection constraints were used in any of the proposed systems (E, KL, ETS, KLTS). This was to fully test the ability of the DNN to 'embed' the required

[2]The linguistic contexts which appeared three times in the training data were from occurrences of sentences beginning with "we aren't".

Table 9.2: *Conditions included in listening test*

ID	Description	Postfilter
N	Natural speech	n/a
V	Vocoded speech	n/a
D	Stacked bottleneck DNN system (Wu et al., 2015)	PF
H	Standard tied HMM system (HTS demo)	GV
F	HMM system w/ fully untied tree (MDL = 0) – variance parameters from system H	PF
CT	Rich context system (Yan et al., 2009) – tri-phone pre-selection	PF
CB	Rich context system (Yan et al., 2009) – bi-phone pre-selection	PF
E	Proposed system w/ Euclidean distance (Section 9.4.3)	PF
KL	Proposed system w/ KLD (Section 9.4.4)	PF
ETS	Proposed system w/ Euclidean distance (Section 9.4.3) – source parameters from system H	PF
KLTS	Proposed system w/ KLD (Section 9.4.4) – source parameters from system H	PF

Table 9.3: *Average candidates per state over a test set*

	CT	CB
overall average	35	196
tri-phone search	29	n/a
bi-phone search	54	193
centre phone search	982	982

Table 9.4: *General setup information on systems*

Training data	2400 sentences (2004 Herald sentences & 396 Harvard sentences) from ‘Nick’ corpus (Cooke et al., 2013a)
Test data	60 Harvard sentences from ‘Nick’ corpus (Cooke et al., 2013a)
Duration model	Natural aligned durations
Number of listeners	30 (each listens to 20 screens)
Testing method	MUSHRA (ITU Recommendation ITU-R BS.1534-1, 2003)

linguistic information into the bottleneck features (context embeddings).

For comparison, a rich-context system guided by a decision tree (rather than the method in Yan et al. (2009) or the proposed method) was created (system F) by growing the decision tree with the MDL factor set to 0. This tree has one leaf per unique context seen in the training data. Variances were borrowed from the standard tied system (H).

2400 sentences from a male speaker of British English were used for training all systems (Cooke et al., 2013a). 60 unseen Harvard sentences were used for testing. STRAIGHT (Kawahara, 2006) was used for speech analysis and the postfilter scaling factor was fixed to 1.2 for all systems (where applied). For all systems, natural durations derived by forced alignment were used. Before presentation to listeners, all utterances were volume normalised (ITU Recommendation ITU-T P.56, 2011). The decision of whether to use a postfilter or GV was made case-by-case for each system, choosing whichever sounded best in informal listening.

9.5.2 Experimental setup

The listening test was conducted using the MUSHRA methodology (ITU Recommendation ITU-R BS.1534-1, 2003), with the same set up as in Chapter 6[3]. As described in Chapter 1, in MUSHRA testing the same sentence is presented to the listener under all conditions, on a single screen. Natural speech is provided as the hidden (i.e., listeners are not told which condition this is) reference and acts as the upper anchor.

30 native English speaking participants with no known hearing impairments were recruited to perform the listening test, with each of the listeners performing judgements on 20 screens (where each screen has the full range of systems synthesising one sentence). Stimuli played to listeners along with listener responses can be found at Merritt et al. (2015c).

[3]The code for conducting the MUSHRA test and analysis of responses was kindly provided by Gustav Eje Henter.

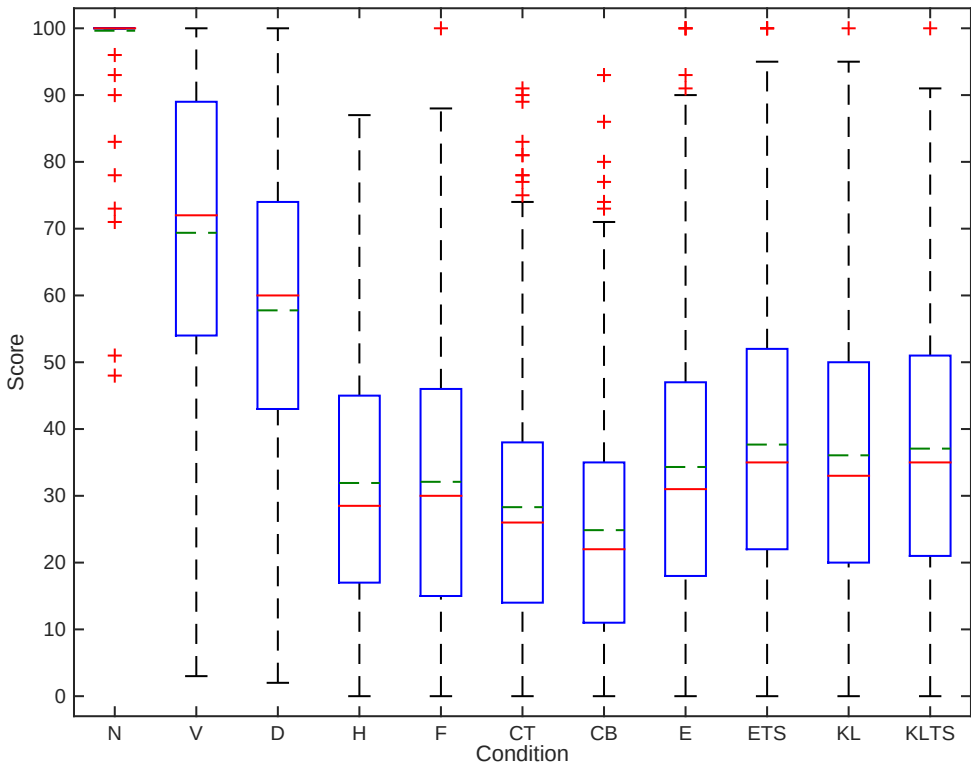


Figure 9.4: *Boxplot of absolute values given from MUSHRA test. Plot uses the same notation as in Figure 4.1. Figure appeared in Merritt et al. (2015b).*

9.6 Results

Listener responses from the MUSHRA test in terms of absolute values awarded to the conditions can be seen in Figure 9.4. All tests for significant differences between conditions applied Holm-Bonferroni correction due to the large number of condition pairs to compare. All conditions are significantly different from all others in absolute rating, except between: H and F, KL and E, KL and ETS, KL and KLTS, ETS and KLTS. Significant differences are in agreement using a t-test or Wilcoxon signed-rank test at a p value of 0.05. These significance tests are described in Chapter 4. The agreement between the t-test or Wilcoxon signed-rank test is illustrated in Figure 9.6. There is a disagreement in statistical significance between conditions F and E: the Wilcoxon signed-rank test finds the difference in judgements to be statistically significant whereas the t-test doesn't. In cases of disagreement between the two statistical significance tests used I interpret this to indicate a small, but significant, difference between the conditions.

Listener responses from the MUSHRA test in terms of rank order awarded to the conditions can be seen in Figure 9.5. All tests for significant differences between con-

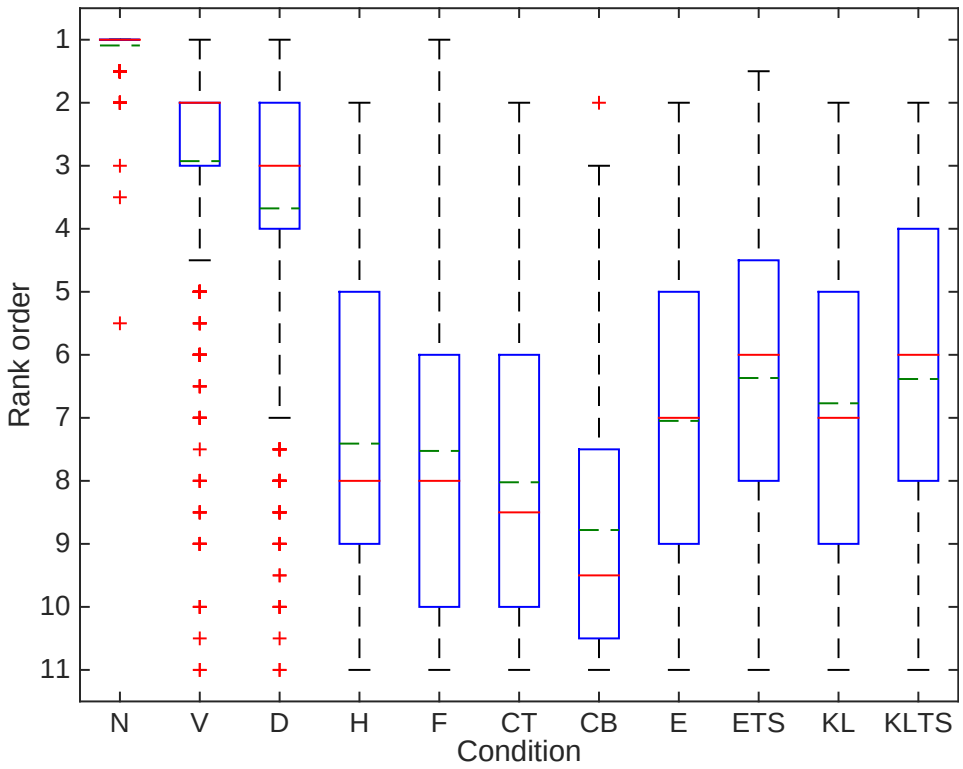


Figure 9.5: *Boxplot of rank order of conditions from MUSHRA test. Plot uses the same notation as in Figure 4.1. Figure appeared in Merritt et al. (2015b).*

ditions applied Holm-Bonferroni correction due to the large number of condition pairs to compare. All conditions are significantly different in rank order except between: H and F, KL and E, ETS and KLTS. These significant differences are in agreement using the Mann-Whitney U test and the Wilcoxon signed-rank test at a p value of 0.05. The Mann-Whitney U test is described in Chapter 6. The agreement between the Mann-Whitney U test and the Wilcoxon signed-rank test are illustrated in Figure 9.7. There is a disagreement in statistical significance between conditions H and E: the Mann-Whitney U test finds the difference in judgements to be statistically significant whereas the Wilcoxon signed-rank test doesn't.

One point of surprise is the ratings given to the CT condition. In informal listening, this was judged to be of higher quality than the H and F conditions. It is possible that the expert listener judgement is out of line with the paid listeners' non-expert opinions on naturalness, a phenomena discussed in Wester et al. (2015). It is suspected that the CT system removes much of the buzzy quality present in system H, but in doing so has made other imperfections audible which are otherwise masked by this buzziness, therefore reducing the perceptual scores from naïve listeners.

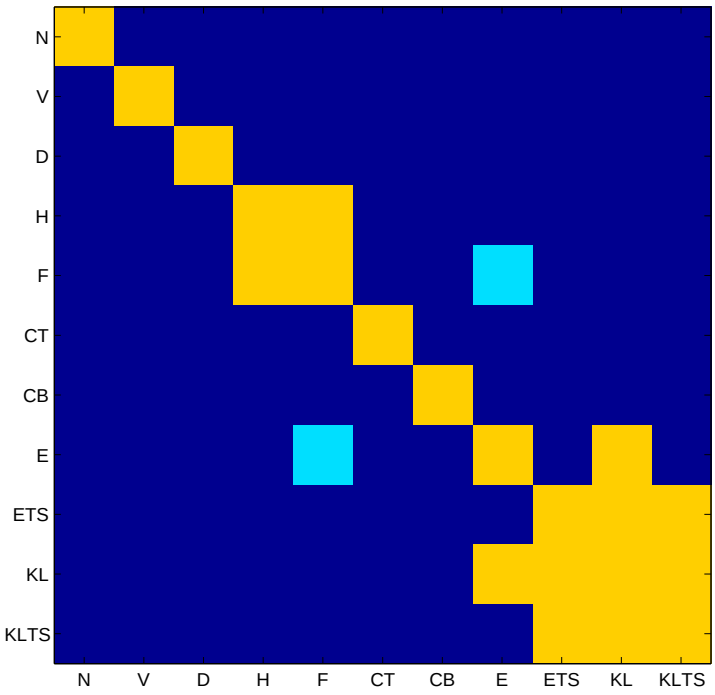


Figure 9.6: *Visualisation of significant differences between systems in terms of absolute value using t-test and the Wilcoxon signed-rank test (p=0.05). Dark blue indicates agreement in significant difference. Yellow indicates agreement in no significant difference. Light blue indicates significant difference found using Wilcoxon signed-rank test but not with t-test.*

No significant improvement in absolute score is observed when source parameters (log fundamental frequency and band aperiodicity) from the standard tied models are used in systems ETS and KLTS compared with KL. This indicates that by performing a KLD search for suitable rich context models we are already incorporating some prosodic information (similar to the level of tied models). The wide range of scores shown on the boxplot for systems V to KLTS shows that this task of scoring these systems is difficult. This is presumably because they are all of quite high naturalness and these variances are caused by different systems being better or worse at differing sentences presented.

The difference in naturalness between systems CT and CB indicate that the preselection implemented in Yan et al. (2009) also steers the system towards selecting better models. This highlights the shortcomings of the reference tied model (system H) used in this system. Conversely, the proposed methods (conditions E, ETS, KL & KLTS), which perform a global search over the training corpus using context embeddings (which embed linguistic context information), allow these systems to select

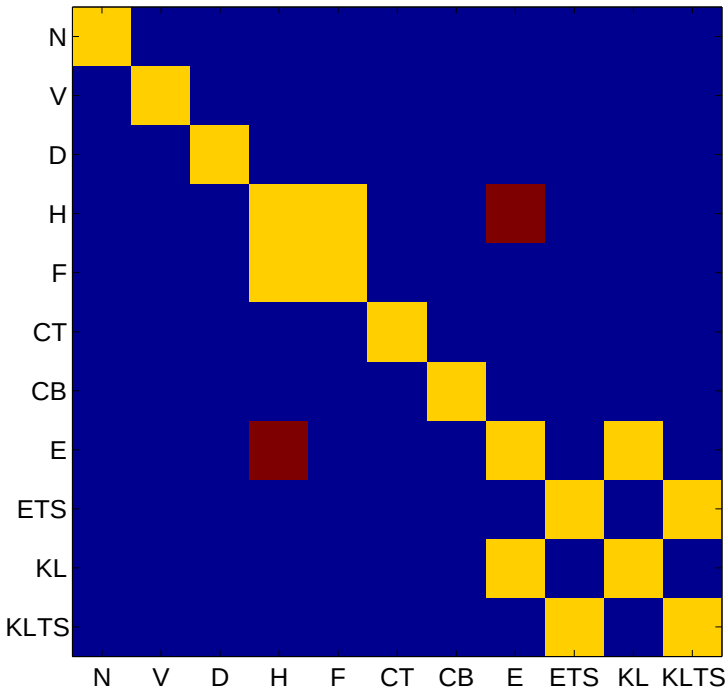


Figure 9.7: *Visualisation of significant differences between systems in terms of rank order using Mann-Whitney U test and the Wilcoxon signed-rank test (p=0.05). Dark blue indicates agreement in significant differences. Yellow indicates agreement in no significant difference. Red indicates significant difference found using Mann-Whitney U test but not with Wilcoxon signed-rank test.*

better rich-context models. This confirms the hypothesis that use of the tied model within HMM synthesis has a large degrading effect on speech quality. Such a finding demonstrates how perceptual testing of hypothesised causes of limited quality in speech synthesis, conducted in Part I of this thesis, is able to reliably inform updates.

Table 9.5 shows how the rich-context models selected in condition KL conformed to various pre-selection criterion. These statistics indicate that the context embedding features used in this investigation are able to steer selection of rich-context models towards similar linguistic contexts, whilst performing a global search of linguistic contexts. This indicates that the search in 'context embedding space' is much more effective than the previous search performed in Yan et al. (2009). Also these statistics give an insight into potential future pre-selection criterion which may be useful for reducing the search space of rich-context models. The search space for conditions E, KL, ETS & KLTS in this investigation is extremely large, with no pre-selection being used in order to not steer the selection of rich-context models. By performing pre-selection which doesn't influence rich-context model selection, the required time to synthesis

Table 9.5: *Conformity of selected rich-context models for condition KL to differing pre-selection criterion.*

Pre-selection criterion	Selected rich-context models which conform
Tri-phone	78.7%
Bi-phone	84.5%
Centre-phone	99.7%

from rich-context synthesis systems can be greatly improved.

Through informal listening, the rich-context systems (systems CT, CB, E, KL, ETS & KLTS) seem to be prone to artefacts at phoneme boundaries, where models from different linguistic contexts meet. The rich-context models are sparsely-trained, high quality models. One hypothesis of the cause of artefacts at phoneme boundaries is that the search thus far of rich-context models has been similar to a greedy target cost function in the unit selection paradigm. This means the search looks for the closest available rich-context model without considering how these models may fit together following MLPG. The introduction of such considerations is something which should be pursued in future work. By considering how rich-context models interact with each other, the potential of the rich-context speech synthesis paradigm, where sparse models are used rather than generalised models, can be fully explored.

The stacked DNN bottleneck synthesis system presented in Wu et al. (2015) outperforms all statistical parametric systems tested in this investigation. This indicates that, while great improvements in quality have been made to HMM synthetic speech, more work is required. It is worth noting that there are many changes in the standard architectures for HMM and DNN synthesis systems, rather than simply a change in the regression method used. This is discussed in Section 11.2. Finally, as already found in the investigations conducted in Part I of the thesis, Figure 9.4 shows that vocoded speech is already significantly less natural than the original waveform.

9.7 Conclusions

The proposed system provides clear improvements over both standard tied HMM models and the previously proposed rich-context model system (Yan et al., 2009). This confirms the hypothesised gains in naturalness from the findings in Part I of the thesis: by moving away from using models calculated by averaging differing linguistic contexts, large gains can be found. This meets the primary goal of the thesis investigation and verifies the proposed simulation framework's effectiveness in identifying real underlying causes of reduced naturalness in speech produced by HMM speech synthesis systems. The findings of the investigation in this chapter also suggest that the previous work on rich-context synthesis did not achieve its full potential. The findings in this chapter indicate that further research into the rich-context synthesis paradigm is required, in order to investigate its full potential. For example, investigation into potential benefits of a measure of suitability for joining rich-context models at synthesis-time would be of interest.

Across the different test sentences used in this investigation the typical state duration was between 2 and 5 frames. When compared with the temporal smoothing effects conducted in Part I of the thesis, the temporal resolution of the state-level generated parameters would appear to be fine for human ears.

Although a state-of-the-art DNN setup is better than all HMM systems here, there is further room for improvement in the HMM systems. This includes the use of an embedding that is specifically designed for the task, not just derived from a DNN that was optimised for synthesis. For example, embeddings could be derived from a DNN that no longer needs to output speech parameters, but perhaps uses more perceptually relevant output features.

The HMM paradigm is much more transparent that the DNN paradigm. Rich-context model parameters can be related directly back to frames in the training data, allowing diagnosis and fault-finding to be carried out. By using the DNN to generate context embedding features which are then used to select rich-context models, we are able to use the unconstrained DNN system to select models for synthesis which have been carefully trained to ensure unsatisfactory characteristics are not present. This link to the training data also suggest simple and obvious ways to build hybrid systems (i.e., statistical model-guided concatenation), which will be investigated in Chapter 10.

9.8 Summary

The rich-context synthesis system introduced in this chapter addresses the following previously highlighted causes of reduced synthesis quality:

- Averaging across differing linguistic contexts has been removed. Instead, only averaging across matching linguistic contexts is performed to produce the mean values of the rich-context models.

- The parameters generated by the rich-context synthesis system contain a much more correct global variance estimation. As a result of this, GV was not used on the trajectories generated by MLPG; only postfiltering is performed.

- While independent modelling of parameter streams is still conducted in terms of the MLPG generation, the context embedding features are however calculated across all parameter streams at the same time. Therefore the embedding features contain information important to all streams. Also, the rich-context models are typically trained only on one instance of a linguistic context and the same group of frames are always present in the rich-context models for each of the different streams. Finally where a selection of rich-context model to be used at synthesis-time is made, this same context model is used for all streams in systems E and KL.

- Given the high level of purity in rich-context models, it is reasonable to predict that trajectories generated from these systems can provide speech which is prosodically much more diverse. Further research in the field of prosody may look to exploit the potential for gains in this area.

9.9 Retrospective review

Whilst the investigation in this chapter was being conducted, inevitably there was further research being conducted within the speech synthesis community. In this section I will comment on work conducted during this time and how it relates to the work in this chapter.

As discussed in Chapter 8, DNNs are now being used to overcome some of the effects of HMM synthesis (Zen and Senior, 2014, Zen, 2015). These systems are found to perform well (Fan et al., 2014, Zen and Sak, 2015, Zen et al., 2013). However it is difficult to investigate the inner workings of the DNN to know whether it is converging upon an undesirable mapping of linguistic contexts. As discussed in the conclusions of this chapter, it is of interest to investigate whether using a DNN to provide selection of models with strict constraints in place (such as rich-context models) is definitively better or worse than relying on Deep Learning.

An alternative approach in the literature for reducing the effects of modelling, is to create a separate model which learns to re-instate the spectral detail lost during modelling. The approach uses a DNN postfilter which restores detail to the spectral domain (rather than the 'model-able' parameters) following parameter generation (Chen et al., 2014). This method was found to be able to outperform GV. However as was discussed in Part I of the thesis, it is expected that applying improvements to the fundamental synthesis approach, as is the case with the rich-context system presented in this chapter, will lead to more substantial improvements to synthesis performance than by applying postfilter fixes to the parameter trajectories produced. However such a postfilter approach would be expected to be complementary to improved modelling.

There has been work reported in the literature using a neural network to generate parameter trajectories, instead of the standard MLPG algorithm (Hashimoto et al., 2015). This initial investigation found that using a neural network to generate the parameter trajectories did not perform as well as MLPG, however this investigation indicates that similar methods may in future be investigated to replace MLPG with a machine learning-based approach.

The investigations in Hu and Ling (2016) and Takaki et al. (2015), look at using neural networks to replace the parametrisation derived from the output from a vocoder. By modelling such an alternative parametrisation of speech it is possible that the issue of clustering differing linguistic contexts, as in decision tree regression, may be reduced. In Chapter 4, alternative parametrisations of speech were investigated and

there was found to be little difference in the effects of modelling between the different parametrisations, although it must be pointed out that the parametrisations investigated in Chapter 4 were derived from the same vocoded spectrum. If a neural network-derived parametrisation of speech was found to lead to improved modelling performance, it is predicted that the improvements in synthesis quality from performing rich-context modelling in this chapter would track those improvements.

As discussed in Section 9.6 of this chapter, there are some inconsistencies as a result of neighbouring rich-context models being poorly matched. One approach discussed in Section 9.6 to overcome this was to consider a form of join cost for rich-context model selection. Another possible reason for the inconsistencies between neighbouring rich-context models may be due to sudden changes in prosody. If this is the case then the use of methods to control the prosody across differing rich-context models would be required. There are a number of investigations in the literature which have looked at using wavelet decomposition to model different levels of prosody within the utterance (Suni et al., 2013, Ribeiro and Clark, 2015, Ribeiro et al., 2015, 2016). Such a decomposition of the parameters output by rich-context synthesis may allow for some of the effects of sudden changes in the prosody between consecutive models to be reduced. For example wavelet decomposition could be used as a postfiltering method to adjust prosodic effects from rich-context synthesis.

Chapter 10

Avoid generation using parametrised speech - hybrid synthesis

This chapter is an expanded version of the work in Merritt et al. (2016a) and therefore the text is closely related to that.

This work was completed in collaboration with others. Discussion of ideas was done between myself, Robert Clark, Junichi Yamagishi, Zhizheng Wu and Simon King. The code for the feed-forward DNN system used was provided by Zhizheng Wu. The code for running and analysing the MUSHRA test was provided by Gustav Eje Henter. The Festival Multisyn hybrid class, used to implement the range of different systems tested, was designed by myself and Robert Clark. This was then implemented into the Festival Multisyn code by Robert Clark. The methods to process the output from the different SPSS systems tested was implemented by myself. Fine-tuning of the systems was then done by myself with the help of Oliver Watts and Zhizheng Wu.

10.1 Motivation

Part I of the thesis investigated a wide range of hypothesised causes of the reduced quality experienced of HMM synthesis. By better understanding what is holding back current synthesis approaches, research effort into improving synthesis systems can be more effective. The investigations found that averaging across differing contexts, as typically occurs while constructing models at the leaf nodes within decision trees, is detrimental to the quality of synthesised speech. Chapter 9 implemented a system which performed better averaging by removing averaging across differing linguistic

contexts and instead only averaging across examples of matching linguistic contexts. Part I of the thesis, however, also found that the parametrisation of speech (i.e., vocoding) introduces a large drop in the quality of speech output, before any modelling has even taken place. Therefore in order to make further gains in synthesis naturalness, systems which remove the effect of vocoding (a standard part of the synthesis pipeline for statistical parametric speech synthesis) should be investigated.

Unit selection synthesis is an obvious example of a system unaffected by vocoding. These systems perform very little processing of the original speech waveforms from the training data and instead aim to join natural sections of speech to produce very natural-sounding speech. In this chapter the use of statistical parametric systems to inform unit selection (hybrid synthesis) will be investigated. Hybrid systems aim to make use of the benefits of both flexible statistical parametric synthesis systems and highly natural unit selection synthesis systems. The starting point of this investigation is a prototypical unit selection system (Festival's Multisyn engine). From the standard Multisyn configuration, the influence of parametric systems on system performance will be observed.

10.2 Prior work

10.2.1 Unit selection

Unit selection synthesis is usually described as an optimisation problem: to find the sequence of units (diphones, in Multisyn) that minimises the sum of target costs and join costs (Hunt and Black, 1996). This involves trading off how well a candidate unit meets a required specification against how well it concatenates with neighbouring units. By defining the join cost to be zero for units that are contiguous in the database, unit selection effectively uses relatively large units of variable size.

Standard unit selection systems typically use mismatches between the linguistic specifications of the target and candidate units to compute a target cost. Distances between acoustic features are used to compute the join cost (Black and Taylor, 1997, Taylor et al., 1998, Taylor, 2006a).

Whilst speech within contiguous regions found in the database is effectively 'perfectly natural', unit selection speech generally suffers from concatenation artefacts. A variety of hybrid synthesis systems have been proposed to solve this problem by employing statistical models to predict the acoustic properties of speech, and then se-

lecting units from the database that best match (Qian et al., 2013, Ling et al., 2008, Yan et al., 2010).

10.2.2 Hybrid synthesis

Hybrid synthesis systems use statistical models (usually by generating speech parameter trajectories) as the basis of the target cost function (Ling et al., 2008, Yan et al., 2010, Qian et al., 2013). An extension of this approach is 'multiform' synthesis in which some types of units are generated via vocoding, whilst others are retrieved from the speech database (Pollet and Breen, 2008, Sorin et al., 2011, 2012, 2014, Fernandez et al., 2015) although this is outside of the scope of the current investigation. Hybrid systems have performed very well in Blizzard Challenges (King and Karaiskos, 2011, 2012, 2013, King, 2014) and are said to make use of the benefits of both the underlying statistical parametric system used and the unit selection system which this drives.

In standard unit selection systems the target cost is typically based on binary distinctions of matching or not matching features between the target and the candidate. Each of the feature-based distinctions, in turn, requires manual tuning of their relative weight to optimise performance. Instead, in hybrid synthesis this target representation is usually within a domain suitable for providing a direct measure of how the candidate matches the target (e.g., speech parameters). Hybrid systems allow for fewer manually-defined rules and weightings, with selection of units being learned from data. Therefore by moving from the standard unit selection target cost function to using SPSS-informed target cost functions (in the hybrid paradigm) may provide a more effective measure of the suitability of a candidate with respect to the target linguistic context.

HMMs are the preferred statistical models in hybrid systems' target cost function, despite recent but compelling evidence that DNNs are superior to the decision tree-based regression employed in standard HMM systems (Wu et al., 2015, Ling et al., 2015, Zen, 2015). An exception to this is the investigation in Fernandez et al. (2015), where a bidirectional recurrent neural network (RNN) provides a prosodic target. However Fernandez et al. did not use this system to synthesise exclusively from the speech database and instead used this in a multiform setup combined with modelled prosody. Previous investigations were also made into the use of RNNs for generating prosodic information for Mandarin (Chen et al., 1998), with diphone synthesis being one of the use-cases discussed for this system. However the details of how this is used to inform

the synthesis system is not made clear and was not formally tested. Since hybrid systems are said to combine the benefits of both systems involved (statistical parametric and unit selection), investigation into the effect of a hybrid system driven by a statistical parametric system with increased naturalness (in generating vocoded speech) is of real interest. The hypothesis is that the increase in statistical parametric speech synthesis naturalness obtained by moving from the standard decision tree-clustered HMM system to the feed-forward DNN system, is also reflected in the naturalness of the resulting hybrid system.

In Chapter 9, context embeddings (which can also be called 'bottleneck features' when they are derived using a hidden layer of a feed-forward neural network (Wu et al., 2015)) were used to select rich-context HMM models (models whose means are trained only on samples where the linguistic contexts exactly match). This involved a search to select models for synthesis in the inevitable event that linguistic contexts encountered at synthesis-time were not observed in the training data (Yan et al., 2009). Models in standard decision tree-based HMM systems are trained by averaging across multiple seen linguistic contexts to account for unseen linguistic contexts in the training data. As rich-context models only perform averaging across examples with matching linguistic contexts, these models are sparsely trained and no longer able to generalise for unseen contexts. Instead, a search for the closest linguistic context present in the training data is required and the corresponding rich-context model is then used for synthesis. The search, when performed in this context embedding space, was found to outperform both conventional HMM synthesis and the previously proposed rich-context model search method, which used the conventional tied-context distribution as a target for the search. This provides a clear motivation to use these context embeddings to select units in a hybrid system, given that the search for closest seen linguistic context is very similar between these two synthesis paradigms. In this investigation, distance in context embedding space (or distance in speech parameter space in some cases) is used to measure the mismatch between the target and a candidate unit.

10.3 Multisyn

Multisyn is a general purpose unit selection framework enabling simple implementation of unit selection synthesis within the Festival toolkit (Clark et al., 2004, 2007, Taylor et al., 1998). Festival's Multisyn is a recognised standard unit selection system and is used as one of the baselines for the Blizzard Challenge. Therefore Multisyn

forms the basis of the hybrid unit selection systems tested in this investigation. The unit size used in all systems reported here is the diphone. Although gains have been demonstrated using other sized units (Qian et al., 2013, for example), this is outside the scope of the current investigation.

The Multisyn target cost function is a simple weighted sum of mismatches in selected linguistic features. The default weights were left unchanged for the baseline system used in this investigation, but the relative weight of the target cost compared to the join cost was manually tuned, for consistency with the hybrid systems to which it was compared. The components of the target cost function are as follows[1]:

- Matching position in phrase.
- Matching stress.
- Matching part of speech tag.
- Matching position in syllable.
- Matching position in word.
- Left phonetic context matches
- Right phonetic context matches
- Bad f_0 (candidate is detected as having incorrect pitch markings)
- Bad duration (candidate duration is unusual - outside 2 standard deviations of the mean duration)

The join cost for Multisyn is a sum of distances between 12 MFCCs, f_0 and energy from the frames either side of the join (Clark et al., 2004, 2007). This default join cost was used in all systems in order to test the effect of the different target costs.

Before performing the search it is necessary to pre-select a shortlist of candidates for each target position. This is to minimise the number of join and target costs required to be computed. The default pre-selection method in Multisyn returns candidates with matching diphone identity. In the event that this list is empty, a back-off scheme is invoked which uses manually-written phone substitution rules. Again, this default scheme was left in place, although it may be possible in future to use distance in context embedding or speech parameter spaces in the pre-selection or backoff procedure.

[1]The bad f_0 and bad duration components of the target cost function are unreported in Clark et al. (2007) however are present in the Multisyn code, presumably being added to the system since the publication.

10.4 Proposed hybrid target cost

The context embeddings derived from a neural network, or alternatively the actual speech parameters predicted at the output of the network, can be thought of as a non-linear projection of the input linguistic features. The projection is learned in a supervised manner, according to whatever optimisation criterion is used to train the network. It is this supervision from acoustic information that makes these DNN-derived features more powerful than the purely linguistic feature-based function used as standard in Multisyn. For example, linguistic features that are not predictive of acoustic properties will be discarded.

The motivation for using a DNN – that, crucially, has been trained to perform parametric speech synthesis – to provide the context embeddings (rather than some other method), comes from the universally positive reports of DNN synthesis in recent literature. The improvements in synthesis quality from moving from standard HMM synthesis to DNN synthesis was shown in Chapter 9.

Multisyn operates on diphone units, but the synthesis DNN we used operates on phone units. To map between these, each phone was divided into 4 uniform sections. The features being used for the target cost (either context embeddings from a DNN (Wu et al., 2015), or output speech parameters from the neural network or a HMM) are gathered together across all frames within each of these 4 regions, from which we compute the mean and variance per section. The variance is floored at 1% of the global variance per feature (the floor value was chosen instead of an arbitrary small value, following informal listening). This is done in the same way for both candidate and target.

The Kullback-Leibler divergence (KLD) (Hershey and Olsen, 2007) is computed for each of the 4 sub-phone regions individually[2]. The use of KLD in context embedding space follows on from the previous work on 'rich-context' modelling in Chapter 9.

The KLD between distribution f of the features computed for the frames corresponding to a given section in the test sentence, and the stored feature distribution g

[2]The Multisyn class for incorporating the distributions of features output from the different statistical parametric systems tested was designed by myself and Robert Clark and implemented into the Multisyn framework by Robert Clark. The processing of features output from the statistical parametric systems into the uniform sections of feature distributions, which are accepted by this Multisyn class, was done by me.

relating to a section of a unit in the unit database, is:

$$D_{KL}(f||g) = \frac{1}{2}[log\frac{|\Sigma_g|}{|\Sigma_f|} + Tr[\Sigma_g^{-1}\Sigma_f] - d$$

$$+(\mu_f - \mu_g)^T\Sigma_g^{-1}(\mu_f - \mu_g)], \quad (10.1)$$

where μ and Σ are mean and covariance and d is the dimensionality of the feature vector. The KLD for each of the 4 sections comprising a diphone is summed together to give the final divergence score. The average of $D_{KL}(f||g)$ and $D_{KL}(g||f)$ was used in order to make the measure symmetrical.

The SPSS-derived target cost function used in this investigation is designed to be independent of phoneme duration, relying on the target cost to select candidates with suitable durations. However, work on explicit control of duration may be fruitful in the future.

10.5 Experiments

10.5.1 Setting target cost weight

For each unit selection configuration to be tested, the target cost needs to be weighted against the join cost in order to optimise synthesis performance. This was done by synthesising a collection of 20 Herald newspaper news sentences which were unseen in the training data. First, all of the target costs calculated in the lattice for these 20 sentences were stored (these all match the pre-selection criterion, therefore represent a spread of reasonable target cost values). The target cost weight was then set so that the distribution of target costs from the lattice search is scaled to be in the range between 0 and 1. Finally weight values surrounding this value were used to generate the development sentences. These were informally tested across listeners who are speech experts. The selected target cost weight value was then used to generate the test sentences used for formal testing.

10.5.2 Implementation

The systems shown in Table 10.1 were constructed in order to test the effectiveness of speech parameter trajectories (from HMMs or DNNs) and context embedding trajectories (from DNNs) for computing the target cost. As previously stated, the only

Table 10.1: *Conditions included in listening test*

ID	Description
N	Natural speech
M	Multisyn
LE	Multisyn with target cost derived from context embedding from 2nd layer of 6 layer DNN (as in Chapter 9)
HE	Multisyn with target cost derived from context embedding from 5th layer of 6 layer DNN
NP	Multisyn with target cost derived from output from Stacked bottleneck DNN system (Wu et al., 2015)
HP	Multisyn with target cost derived from output from HTS demo with GV (Zen et al., 2007a)

Table 10.2: *General setup information on systems*

Training data	2400 Herald (2004) & Harvard (396) sentences from 'Nick' corpus (Cooke et al., 2013a)
Test data	90 Herald sentences
Duration model	Predicted by HTS (Zen et al., 2007a)
Number of listeners	30 (each listens to 30 screens)
Testing method	MUSHRA (ITU Recommendation ITU-R BS.1534-1, 2003)
Join cost	Pitch, energy & spectral mismatch between either side of join (Clark et al., 2007)
Pre-selection criteria	Matching diphone (Clark et al., 2007)
Back-off rules	Manually-defined diphone substitution rules (Clark et al., 2007)

component that differs between systems is the target cost, and the relative weight between target and join costs (tuned as described in Section 10.5.1). The general system configurations across all systems are shown in Table 10.2.

Systems LE and HE use 32-dimensional context embedding features generated by a DNN similar to that described in Wu et al. (2015) and Wu and King (2015). These come from the 2nd (lower layer of the DNN, closer to the linguistic input) or 5th (higher layer of the DNN, closer to the speech parameters output) layer of a 6 hidden layer feed forward DNN, respectively. Different layers of the DNN were

used to create context embeddings to test whether embeddings generated closer to the linguistic input or to the speech parameters output from the DNN affect unit selection performance. These systems are successors to the rich context system described in Chapter 9 but instead of generating the speech using a vocoder, they perform unit selection and concatenation.

System NP uses the speech parameters output from the final layer of the stacked bottleneck DNN system presented in Wu et al. (2015) and Wu and King (2015). The speech parameters form an 86-dimension vector (60th order Mel-generalised cepstrum, 25 band-aperiodicities, f_0), following MLPG.

System HP was included to represent a conventional HMM-guided hybrid system. This system uses the parameters generated by HMMs trained using the HTS demo recipe (Zen et al., 2007a), including GV, to compute the target cost in much the same way as system NP makes use of the generated speech parameters from the stacked bottleneck DNN system. The dimensionality of the speech parameters generated from the HMM system matches that generated by the stacked bottleneck DNN system (60th order Mel-generalised cepstrum, 25 band-aperiodicities, f_0). The comparison between these systems HP and NP will tell us whether the gains offered by DNNs in SPSS carry over to the hybrid scenario.

Informal listening to the speech generated via vocoding from the speech parameters of systems NP and HP was conducted, in order to confirm that these systems generate speech of the quality expected. This vocoder-generated speech was not evaluated formally in the listening test reported below. The relative target cost vs join cost weight for all systems (M, LE, HE, NP and HP) was tuned by informal listening, as discussed in Section 10.5.1.

2400 sentences from a male speaker of British English (Cooke et al., 2013a) were used as the training set for the HMMs and DNNs, and as the unit database in all systems. The text of 20 unseen Herald newspaper news sentences was used as a development set for tuning the target cost weight of each system. An additional 90 unseen Herald news sentences were then used for the listening test.

A larger number of sentences were used for testing in this investigation than in Chapter 9 as unit selection systems are more prone to sentence-specific judgements due to the nature of using unmodified units of speech. These result in less stable speech than is produced by statistical parametric synthesis systems. By testing over a larger number of sentences the effects of these sentence-specific judgements are mitigated. Unseen Herald sentences were used rather than the Harvard test sentences

from the 'Nick' corpus (Cooke et al., 2013a), as the Harvard sentences were found to have more rare diphones present which may result in an unreasonably degraded performance. There is also a larger number of Herald sentences present in the training set than Harvard sentences which is why these were selected for the test sentences. There were no instances of missing diphones for the test sentences used, resulting in no use of the manually-defined back-off rules. Additional recordings of the same speaker for these sentences were used for system N in the listening test.

For DNN synthesis (required to produce the embeddings or speech parameters of systems LE, HE and NP), durations predicted by the HMM system were used; note that the durations of the final hybrid synthetic speech are determined by the unmodified natural durations of the candidate diphone units selected from the database. Before conducting the listening test, all utterances were volume normalised according to ITU Recommendation ITU-T P.56 (2011).

10.5.3 Experimental setup

The listening test followed the MUSHRA paradigm (ITU Recommendation ITU-R BS.1534-1, 2003), comprising the systems shown in Table 10.1. System N acts as the hidden (i.e., not labelled in the test) upper anchor. This test was conducted with 30 native English speaking participants with no known hearing impairments. Each listener rates 30 screens. Each screen presented 6 stimuli at once: a single sentence under all 6 conditions. The 30 listeners were split into 3 groups of 10 listeners. Each group of listeners was presented with a disjoint set of 30 sentences; thus 90 different sentences were used. The stimuli played to listeners along with listener responses can be found at Merritt et al. (2016b).

10.6 Results

Listener responses from the MUSHRA test in terms of the absolute values of their scores are shown in Figure 10.1. All tests for significant differences used Holm-Bonferroni correction due to the large number of condition pairs to compare. All conditions are significantly different from each other in terms of absolute value, except between: M and HP, LE and HE, LE and NP, HE and NP. Significant differences are in agreement using a t-test and Wilcoxon signed-rank test at a p value of 0.01. These significance tests are described in Chapter 4. The agreement between these tests is

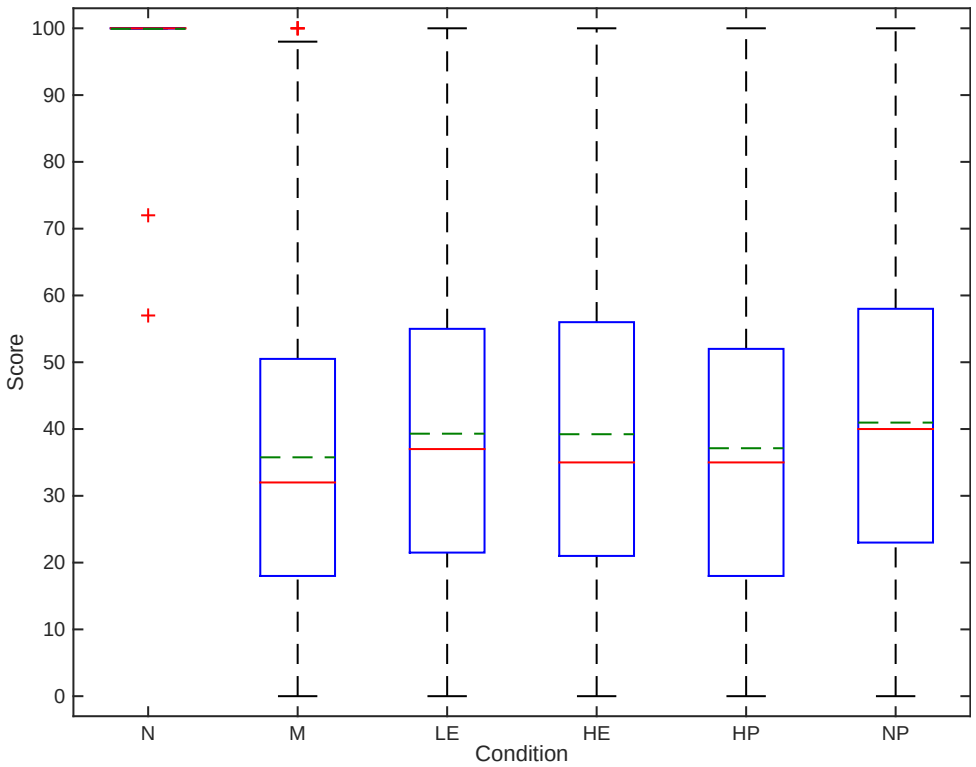


Figure 10.1: *Boxplot of absolute scores from MUSHRA test. Plot uses the same notation as in Figure 4.1. Figure appeared in Merritt et al. (2016a).*

illustrated in Figure 10.3.

Listener responses from the MUSHRA test in terms of the rank ordering of systems derived from their scores are shown in Figure 10.2. All tests for significant differences used Holm-Bonferroni correction due to the large number of condition pairs to compare. All conditions are significantly different from each other in terms of rank order, except between; M and HP, LE and HE. These significant differences are in agreement using a Mann-Whitney U test and a Wilcoxon signed-rank test at a p value of 0.01. The Mann-Whitney U test is described in Chapter 6. The agreement between these tests is illustrated in Figure 10.4. There is a disagreement in statistical significance between conditions LE and NP with the Mann-Whitney U test finding this difference in ranking to be statistically significant whereas the Wilcoxon signed-rank test does not.

10.6.1 Comparison to baseline system M

We can see that the 'trajectory tiling' approach to unit selection described in Qian et al. (2013), and implemented in our systems LE, HE, HP, NP is generally effective, with all systems performing at least as well as the baseline, and significantly better in all

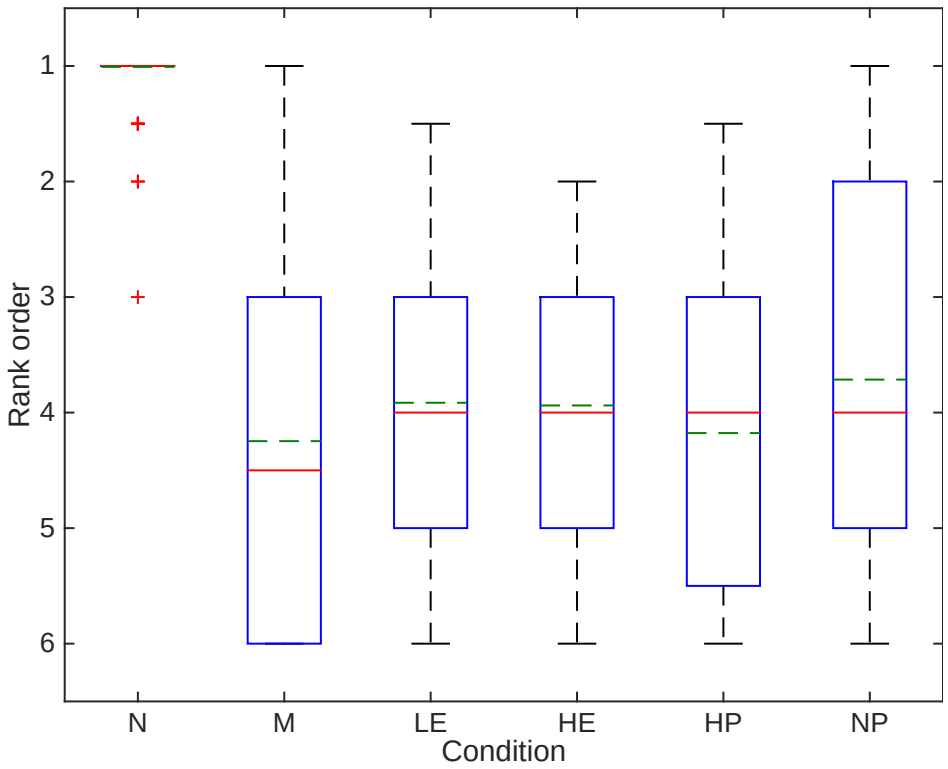


Figure 10.2: *Boxplot of the rank order from MUSHRA test. Plot uses the same notation as in Figure 4.1. Figure appeared in Merritt et al. (2016a).*

cases where a DNN was used as the parametric model. We were not able to obtain significant improvements over the baseline with HMM-generated speech parameter trajectories (system HP).

10.6.2 DNNs vs HMMs

The use of deep neural networks in systems LE, HE and NP provides significant improvements over both the baseline (M) and the HMM-driven hybrid system (HP). This demonstrates that the gains found in SPSS systems, when moving from decision tree-clustered HMMs to DNNs, transfers over to the hybrid unit selection paradigm.

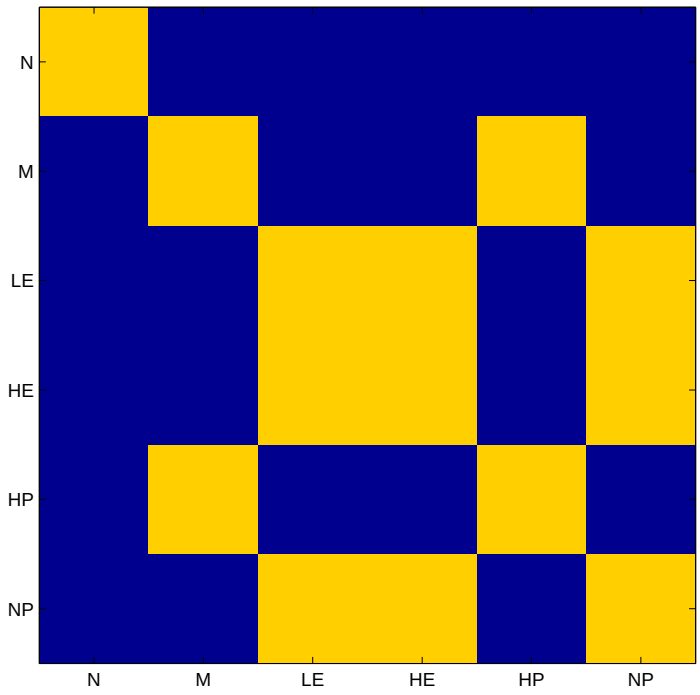


Figure 10.3: *Visualisation of significant differences between systems in terms of absolute value using t-test and the Wilcoxon signed-rank test (p=0.01). Dark blue indicates agreement in significant difference. Yellow indicates agreement in no significant difference.*

10.7 Conclusions

In this chapter the use of deep neural networks to guide unit selection systems has been proposed. An experimental comparison of several different configurations of hybrid unit selection has also been presented. These were all implemented within Festival's Multisyn framework. The use of a DNN to generate features for use in the target cost was found to be more effective than using a HMM, be that using the speech parameters generated at the output of the DNN or using context embeddings from a bottleneck layer.

In this investigation, only the target cost function was modified. However, further increases might be obtained in future work by improving the join cost function. This might be done by introducing a search for the optimal join position (Conkie and Isard, 1997), or calculating the join cost using an alternative representation (Richmond and King, 2016, Vepa and King, 2006).

Although no significant differences were found between the use of speech parameters from a DNN compared to context embeddings, there is perhaps more consis-

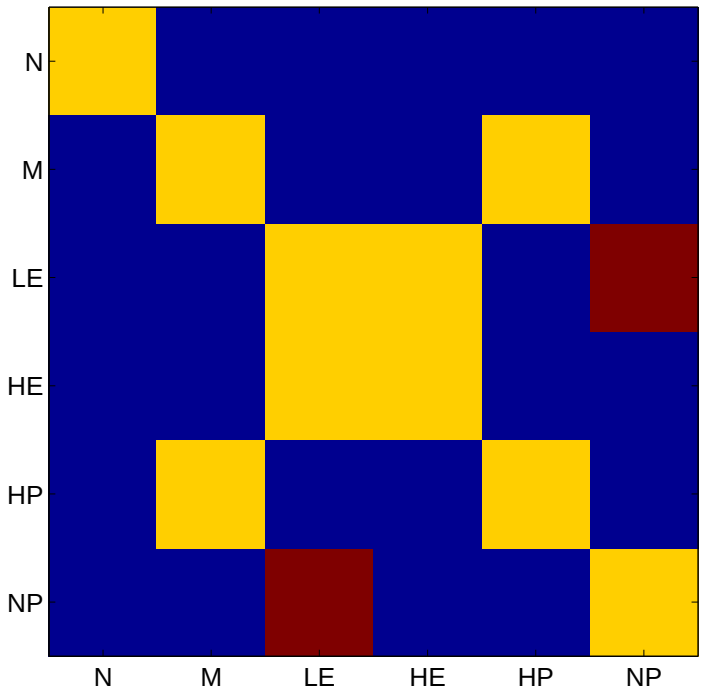


Figure 10.4: *Visualisation of significant differences between systems in terms of rank order using Mann-Whitney U test and the Wilcoxon signed-rank test (p=0.01). Dark blue indicates agreement in significant differences. Yellow indicates agreement in no significant difference. Red indicates significant difference found using Mann-Whitney U test but not with Wilcoxon signed-rank test.*

tency in listener judgements for the embedding-based systems (LE, HE) than the DNN speech parameter-based system (NP). This can be observed in Figure 10.2.

The context embedding features discussed here could be used elsewhere in the unit selection system. For example these features may be incorporated into the back-off function, replacing manual phone substitution rules, or used to perform the initial pre-selection of units, instead of the current pre-selection of units by matching diphone identity.

Investigating different types of neural network for use in this hybrid framework is left as future work. But it is expected that any improvement in parametric synthesis would carry over to the hybrid method for waveform generation, given the performance increase that was noted in this investigation from moving from the standard HMM system to a DNN system for driving the target cost function. For example, mixture density networks (MDNs) might be used to directly produce a likelihood-based target cost instead of the KLD-based approach used here. RNNs, which are more powerful sequence models, might also be used to generate the target trajectories.

The systems investigated here made use of uniform subsections of diphones (the unit size) in conjunction with phoneme-level acoustic models. However investigations into using diphone-level acoustic models is of future interest as this would allow state-sized representations of speech to be used instead and may further improve performance.

Investigation into the use of different speech coding methods, such as Stylianou (2001), may be of future interest in order to improve joins between units.

Direct comparisons between SPSS and exemplar-based systems is very challenging, as the performance of an exemplar-based synthesis system is extremely dependent on the size of the speech corpus available. The corpus used in this investigation is a substantial corpus for performing SPSS, however for the unit selection and hybrid systems investigated in this chapter the number of utterances in the corpus is quite low. Therefore it was not deemed that the inclusion of SPSS systems in the tests conducted in this investigation would be very enlightening. An extended version of the hybrid system introduced here was entered into the Blizzard Challenge (Merritt et al., 2016c), which made use of much larger amounts of training data than was the case for the investigation in this chapter. The perceptual testing of the challenge involved SPSS and unit selection systems from other participating teams. The hybrid system entered into the challenge was found to perform very well against such systems. As hybrid synthesis appears to inherit some of the problems of unit selection synthesis, in terms of data sparsity, investigation of differing unit sizes and signal modification at unit joins, which were both out of the scope of the current investigation, are of interest in order to improve performance where a reduced amount of training data is available.

Hybrid synthesis makes use of both unit selection and SPSS systems. In Part I of the thesis the limitations of SPSS were investigated in order to gain knowledge of what causes reduced synthesis quality. The same style of perceptual testing in order to provide informed improvements could be applied to the unit selection aspect of hybrid synthesis. Although such an investigation is outside the scope of the thesis, it is believed that this would lead to further improvements in the quality of synthesised speech.

10.8 Summary

The hybrid speech synthesis system introduced in this chapter addresses a number of causes of reduced synthesis quality highlighted in Part I of the thesis. The solution to these issues are as follows:

- Vocoding is no longer present in the synthesised speech, resulting in much higher quality being possible.
- Speech is no longer generated from models, instead real examples (units) of speech are used. Statistical parametric models are used to guide the selection of these units. As a result of this, causes of degraded speech discovered in Part I of the thesis are no longer present in the speech output. These include: correct variance of speech parameters, independent modelling of parameter streams, diagonal covariance between spectral parameters and averaging across differing linguistic contexts. However it must be noted that some of these different effects may play some part in the SPSS system used to guide the selection of units. Future research into whether there are any implications to the naturalness of hybrid synthesis as a result of the limitations of the underlying SPSS system, may be of interest.

10.9 Retrospective review

Whilst the investigation in this chapter was being conducted, there was further research being conducted within the speech synthesis community. In this section I will comment on work conducted during this time and how it relates to the work in this chapter.

An alternative approach to avoiding the use of vocoding, proposed in the literature, is to directly model the speech waveform (Tokuda and Zen, 2015, 2016). As discussed in Chapter 1, the waveform contains many combined components of speech and as such direct modelling is a very complex task. Research in direct modelling of the speech waveform is still in its infancy.

Another alternative to using a vocoder to extract speech parameters for modelling, investigated in the literature, is to use a neural network to perform the parameter extraction (Hu and Ling, 2016, Takaki et al., 2015). The parameters output by the neural network are then modelled. For generation, the predicted parameters are passed back through the neural network to produce the speech waveform. As is the case with speech waveform modelling, this synthesis approach is still in its infancy.

Chapter 11

Conclusions & future work

11.1 Contribution to future speech synthesis systems

Statistical parametric speech synthesis (SPSS) commonly falls short of the quality of natural speech or unit selection under 'best case' conditions. SPSS however is considerably more flexible than unit selection synthesis and the speech produced has more consistent performance, making research into improvements in this synthesis domain very desirable. Before work on this thesis began, there were a number of hypothesised causes within the literature as to the reduced quality of synthesised speech from SPSS systems. However these hypothesised causes were rarely formally tested, resulting in uncertainty as to what was causing the perceived drop in synthesis performance and as to which causes were causing the largest degradation in synthesised speech.

Investigative methodology

Part I of this thesis therefore presented a methodical approach for confirming or rejecting hypothesised causes of the reduced quality in hidden Markov model (HMM) speech synthesis. In addition to confirming hypothesised causes of reduced synthesis quality, the contribution of found causes towards the overall synthesis quality was quantified. The investigation in this thesis was conducted by placing these hypothesised causes of reduced quality into a conceptual 'continuum of speech'. This continuum ranges from natural speech at one end, to full SPSS at the other end. Each element within the continuum represents a different hypothesised cause of reduced quality. Perceptual findings from this methodology allow for informed and targeted research in speech synthesis. The approach taken in this thesis to confirm or reject hypothesised

causes of degradations via perceptual testing, in order to apply solutions in an informed way, is an approach which can and should be used to approach future problems. Such an approach ensures that any improvements implemented are actually worthwhile and are overcoming *fundamental* causes of reduced synthesis quality. Part I of the thesis highlighted that **averaging across differing lingusitic contexts**, as is performed as a result of decision tree regression in HMM synthesis, and the **parametrisation of speech**, i.e., vocoding, both dramatically degrade the quality of synthesised speech.

Avoid performing averaging across differing linguistic contexts

Following the perceptual testing undertaken, a further contribution of this thesis was the revisiting of the rich-context HMM synthesis paradigm in Part II of the thesis. Given the findings in Part I of the thesis, it was correctly anticipated that this synthesis paradigm had greater potential than was found using its original configuration in Yan et al. (2009). In Yan et al. (2009), standard tied HMM models are used as the target from which to search for the closest rich-context models. Chapter 9 looked to replace the target for rich-context model selection because tied HMM models are created by averaging across differing linguistic contexts. Instead, the use of a reduced dimensionality space to perform the rich-context model search within, was investigated. A search for rich-context models in a lingustic context embedding-space, created by using the activations at a bottleneck layer in a feed-forward DNN, was found to perform significantly better. This improvement is as a result of the rich-context model search no longer being affected by the distortions present in a model calculated by averaging across differing linguistic contexts, as was the case previously with the rich-context model search in Yan et al. (2009). This flaw in the previous rich-context model search was identified as a direct result of the investigation performed in Part I of the thesis. The findings of the investigation in Chapter 9 highlight that further research into the rich-context HMM synthesis paradigm is of interest. This is because rich-context synthesis is still in its infancy, presumably due to the effects the previous target for rich-context model search (tied HMMs) had on subsequent synthesis quality. As discussed in Chapter 9, there are still open questions in rich-context synthesis research. For example, whether the use of a lattice search (like that used in unit selection) to optimise not only how well a rich-context model matches the 'target', but also how well neighbouring rich-context models will join.

Avoid generation using parametrised speech

An additional contribution of this thesis is the investigation of hybrid synthesis in Part II. This investigation was motivated by Part I of the thesis finding that vocoding has a large degrading impact on synthesis performance. Unit selection is an obvious example of a system unaffected by vocoding, however, as previously stated, this synthesis paradigm is not as flexible as SPSS and synthesis quality is less stable. Hybrid synthesis systems are said to incorporate the benefits of the extremely natural underlying unit selection synthesis, with the benefits of the SPSS system used to drive the selection of units. At this point in time it is widely accepted in the literature that gains in naturalness are to be found from moving from the standard HMM synthesis using tied models, to using deep neural networks (DNNs). However at the time this work was conducted, DNN systems were infrequently incorporated in hybrid synthesis systems. Hybrid synthesis is expected to benefit from both the unit selection and SPSS systems being used. Therefore, it was of clear interest to investigate whether changing the SPSS system, used to select units, to an SPSS system which performs better when producing speech in the SPSS domain, results in improved hybrid synthesis performance. The investigation in Chapter 10 found that hybrid synthesis does indeed appear to benefit from both the high quality of unit selection and the flexibility of the underlying statistical parametric system used to guide the selection of units. As a result of this, by improving the SPSS system, used to drive unit selection, to a system which performs better in the SPSS domain does indeed lead to improved hybrid synthesis performance. This finding is significant because it indicates that improvements found in the field of SPSS can also provide benefits to hybrid systems, suggesting that future improvements in SPSS can be expected to also benefit hybrid synthesis. Linguistic context embeddings were investigated in Chapter 10 and found to perform well as the target cost function. However this finding also presents future research questions for the field of hybrid synthesis: this compact representation may be applicable to more than just the target cost function. For example, this reduced dimensionality context embedding space may prove useful to replace the, currently manually-defined, pre-selection or back-off functions within the underlying unit selection system. Such further integration of SPSS within hybrid synthesis may result in further improvements, including the flexibility and stability of synthesis performance.

The focus of the research undertaken within this thesis has been on the effect of modelling on filter parameters. The effect of modelling with respect to prosody is left

as future work, as prosody is a complete field of research in itself. This exemption of prosody has been by design in order to avoid the large number of experimental variations which this extra element requires as well as avoiding the underlying semantics of speech which this also entails. However it is possible that future researchers may be interested in repeating the investigations conducted in this thesis, focusing instead on the effects modelling has on the prosody of synthetic speech.

11.2 Where do the improvements come from when moving from HMMs to DNNs?

In this section the investigation reported in Watts et al. (2016a) is summarised and discussed. My contribution to this work was as part of the initial discussion, from which this investigation took place. The systems included in testing were implemented by Oliver Watts. The perceptual testing was created and analysed by Gustav Eje Henter. The published version of the paper was predominantly written by Oliver Watts; this section has been rewritten in my own words.

While it is widely claimed in the literature (Zen, 2015) that moving from HMM-based systems to DNN-based systems for SPSS results in better synthesis quality, it is often overlooked that the standard setups of these two systems are very different. More than just the underlying regression model has changed. It is therefore unknown exactly which elements of the standard DNN system design is outperforming that of the standard HMM system.

A number of differences in the standard system setups for HMM and DNN systems exist, these will now be described.

- The regression model; a decision tree is used as the regression model in HMM synthesis from which the HMM models are produced, whereas in DNN synthesis the regression model is the neural network. Decision tree regression was found to degrade synthesis performance in Chapters 4 & 9, due to averaging across differing linguistic contexts.

- The granularity of output in the standard system setup also differs: the output from HMM synthesis systems is typically state-level whereas in DNN synthesis this is typically frame-level. However in both HMM and DNN synthesis, MLPG is still used as standard to produce the final smooth speech parameter trajectories.

Table 11.1: *Summary of systems evaluated; V denotes vocoded natural speech.*

System	Regression model	Regression target unit	Stream modelling	Variance	Duration-derived features	Enhancement method
V	-	-	-	-	-	-
D1	decision tree	state	separate	context-dependent	no	GV
D2	decision tree	state	separate	context-dependent	no	postfilter
N1	neural network	state	separate	context-dependent	no	postfilter
N2	neural network	state	separate	fixed	no	postfilter
N3	neural network	state	combined	fixed	no	postfilter
N4	neural network	frame	separate	fixed	no	postfilter
N5	neural network	frame	combined	fixed	no	postfilter
N6	neural network	frame	combined	fixed	yes	postfilter

- The dependence between streams included in modelling also differs between standard setups: stream-independent modelling is the standard in HMM synthesis whereas combined speech parameter estimation is standard in DNN synthesis.

- The way variance for MLPG is calculated also differs between HMM and DNN standard configurations: variance is typically calculated on a context-dependent level in HMM synthesis whereas a fixed value is normally used in DNN synthesis.

- The inclusion of duration-derived features (i.e., an input feature which acts as a counter within the current state) is included in DNN systems whereas this isn't typically used in HMM synthesis.

- Finally GV has long been the standard enhancement method applied in HMM synthesis whereas postfiltering is typically applied in DNN synthesis.

As there are a large number of configuration differences between the 'established' standard setups for HMM and DNN synthesis systems in literature it is of interest to observe the contributing factors of these differences towards synthesis performance. Therefore a number of intermediate configurations between these standard setups are included in subjective evaluations. The range of system configurations tested is shown in Table 11.1.

11.2.1 Evaluation

The contributions of each of the various differences in configurations between standard HMM and DNN system setups were tested using the MUSHRA testing paradigm (ITU Recommendation ITU-R BS.1534-1, 2003). As discussed in Chapter 1, MUSHRA testing usually uses natural speech as an hidden (listeners are not informed which slider corresponds to this condition) upper-anchor for the range of conditions. However due to the large number of conditions present in this investigation it was decided to use natural vocoded speech (natural speech vocoded and parametrised to the speech parameters suitable for modelling) as the upper-anchor instead of natural speech in order to reduce the number of conditions present in the MUSHRA test. Vocoded speech still acts as the upper-bound for conditions in this listening test, however, as this is the parametrisation of speech which is then modelled by the SPSS systems tested. Listeners were instructed to rate the vocoded speech condition (condition V) at 100.

20 listeners were recruited to take part in the listening test with each listener rating two sets of ten Harvard sentences, with each of these sets being approximately phonetically balanced. Seven sets were used in total across all listeners (70 different sentences in total), with each set being presented to at least five listeners and at most to six listeners. The testing stimuli and listener response data for this investigation can be found at Watts et al. (2016b).

11.2.2 Results

The listener responses from the MUSHRA test in terms of absolute values of the scores given are shown in Figure 11.1. All tests for significant differences between conditions applied Holm-Bonferroni correction due to the large number of condition pairs to compare. All conditions were found to be significantly different from all others in terms of absolute rating, except between: D1 and D2, N1 and N2, N1 and N3, N2 and N3, N4 and N5. Significant differences are in agreement using a t-test and Wilcoxon signed-rank test at a p value of 0.05. These significance tests are described in Chapter 4. The agreement between the t-test and Wilcoxon signed-rank test is illustrated in Figure 11.3.

The listener responses from the MUSHRA test in terms of the rank order awarded to the conditions can be seen in Figure 11.2. All tests for significant differences between conditions applied Holm-Bonferroni correction due to the large number of condition pairs to compare. All conditions were found to be significantly different from

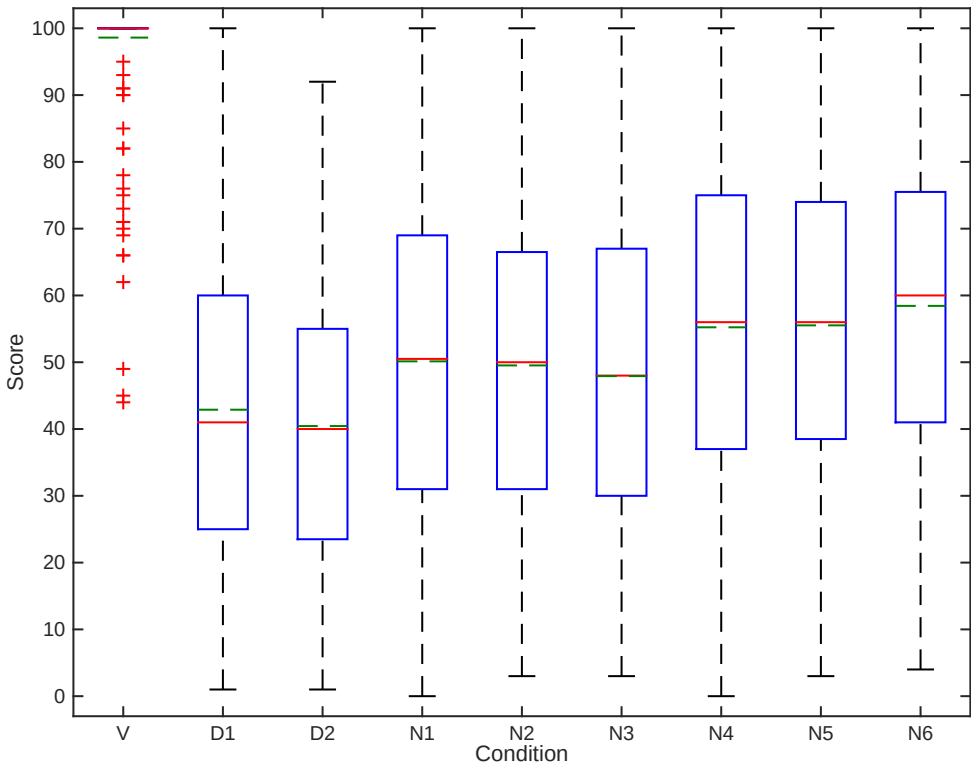


Figure 11.1: *Boxplot of absolute scores from the MUSHRA test. Plot uses the same notation as in Figure 4.1. Figure appeared in Watts et al. (2016a).*

all others in terms of the rank ordering awarded, except between: N1 and N2, N1 and N3, N2 and N3, N4 and N5. Significant differences are in agreement using the Mann-Whitney U test and the Wilcoxon signed-rank test at a p value of 0.05. The Mann-Whitney U test is described in Chapter 6. The agreement between the Mann-Whitney U test and the Wilcoxon signed-rank test is illustrated in Figure 11.4. There is a disagreement in statistical significance between conditions D1 and D2: the Wilcoxon signed-rank test finds the difference in terms of rank order awarded to be significant whereas the Mann-Whitney U test does not.

The listener responses from the MUSHRA test confirm the hypothesis that there are many elements between the standard system configurations of HMM and DNN synthesis systems which contribute to the reported improvements in quality, rather than improvements simply being attributable to switching the regression model from decision trees to neural networks. This is evident by the groupings of system configurations in terms of the naturalness scorings awarded by listeners. Listeners judged there to be gains in naturalness introduced by: switching from decision tree regression-derived models (which are calculated by averaging across differing linguistic contexts) to us-

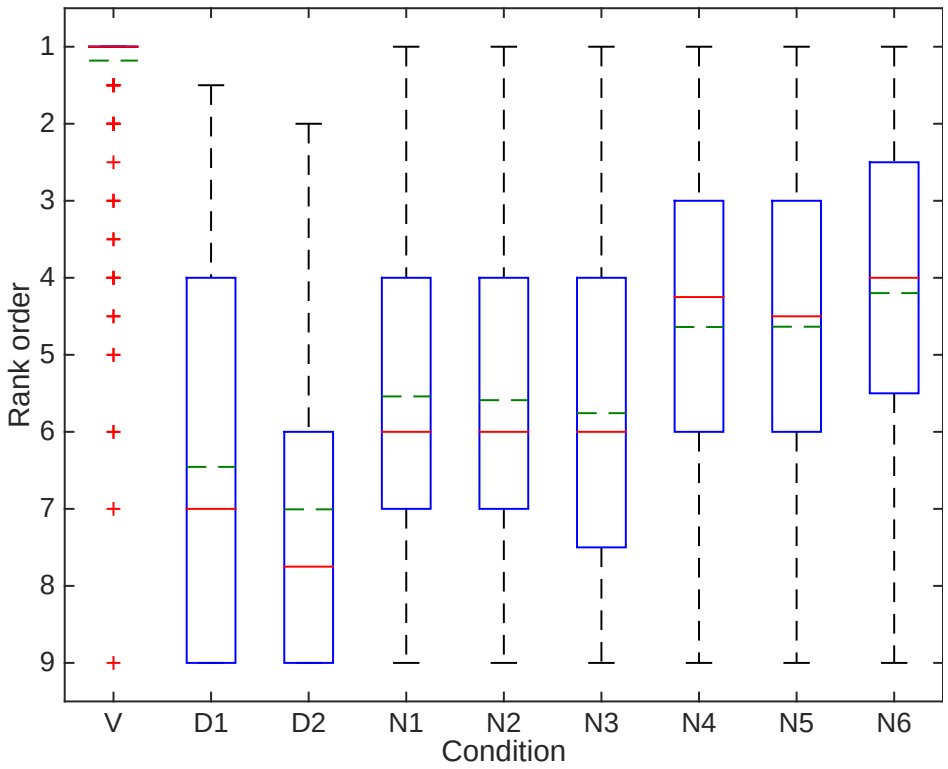


Figure 11.2: *Boxplot of rank order of conditions from MUSHRA test. Plot uses the same notation as in Figure 4.1.*

ing neural network regression (moving from systems D1 and D2 to systems N1, N2 and N3); improved granularity of speech output by moving from state-level output to frame-level output (moving from systems N1, N2 and N3 to systems N4 and N5); and from using duration-derived linguistic features (moving from systems N4 and N5 to system N6) although the magnitude of this increase in naturalness rating is less than between the previous two groups of systems. Interestingly, listeners judged there to be no difference between using context-dependent variance and using a fixed variance in maximum likelihood parameter generation (MLPG), indicating that the simpler approach of using a single variance value to represent how speech parameters vary over time is sufficient for MLPG to perform well. Another point of interest is that listeners judged there to be no difference between combined stream modelling and seperate stream modelling (systems N2 and N3).

One final comment on the findings of this investigation are that it is currently not obvious how much duration-derived linguistic features, which were found to provide gains in the naturalness of output speech, would help if the durations were predicted rather than natural, as was the case in this investigation. The use of natural dura-

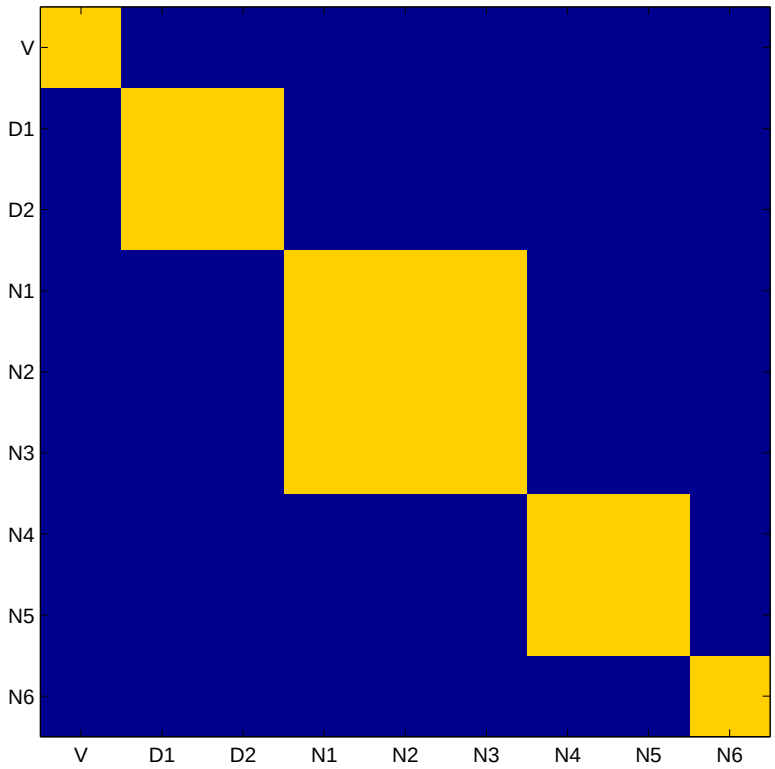


Figure 11.3: *Visualisation of significant differences between systems in terms of absolute value using t-test and the Wilcoxon signed-rank test (p=0.05). Dark blue indicates agreement in significant difference. Yellow indicates agreement in no significant difference.*

tions may unfairly increase the naturalness of the output speech in a way which may be unachievable with the use of current duration models. In fact, it may be the case that the use of current duration models to generate these duration-derived linguistic features may have a detrimental effect on speech naturalness where durations are incorrectly calculated. However the listener responses reported in this investigation do act as an indication of the level of naturalness achievable given a 'perfect' duration model. Future work to investigate the effect of predicted durations on the naturalness of synthesised speech may be of interest.

Due to the large number of conditions present in the evaluation, it was not possible to include many other system configurations which would may have also been of interest. For example inclusion of a HMM synthesis system which produces frame-level output, such as that of Black (2006), would have been of interest in order to see whether this leads to a similar level of improvements in synthesis naturalness as was observed in the DNN synthesis systems. Also the inclusion of the rich-context HMM

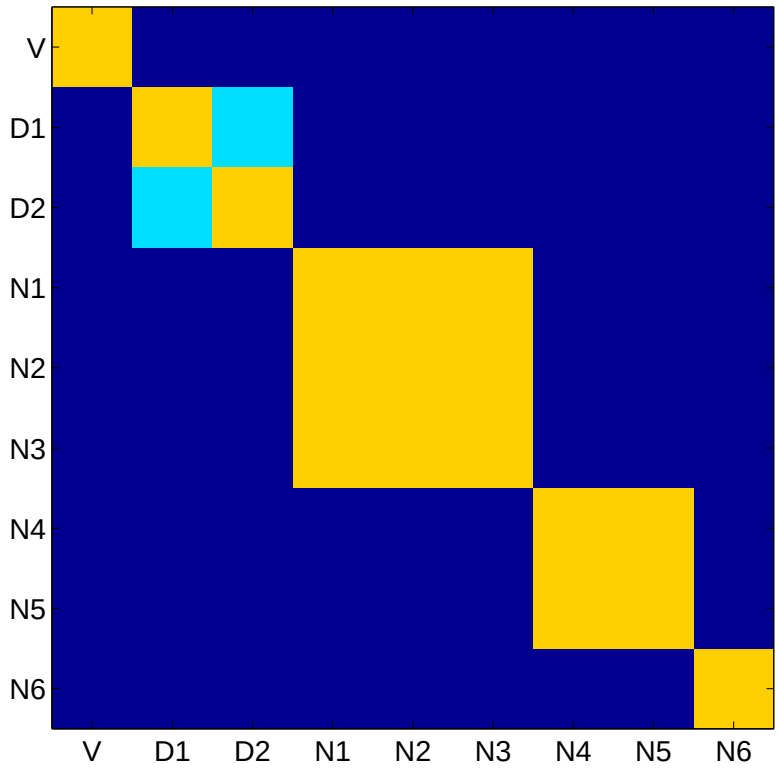


Figure 11.4: *Visualisation of significant differences between systems in terms of rank order using Mann-Whitney U test and the Wilcoxon signed-rank test (p=0.05). Dark blue indicates agreement in significant differences. Yellow indicates agreement in no significant difference. Light blue indicates significant difference found using Wilcoxon signed-rank test but not with Mann-Whitney U test.*

synthesis system introduced in Chapter 9 would be of interest, as this combines HMM and DNN systems. Inclusion of more HMM-based systems may have been of interest in order to observe whether similar system configuration changes lead to similar increases in synthesis performance as for DNN-based systems, or if the improvements in performance are simply as a result of better regression. This investigation has raised interesting observations as to the wholesale changes in system architecture between HMM and DNN synthesis systems; further development of this line of investigation is left as future work.

11.3 Parametric vs time domain representation

As discussed in Section 11.2, the improvements in synthesis performance reported in the literature from moving from HMM to DNN systems, is attributable to multiple factors rather than simply being as a result of altering the regression model. For example, the rich-context synthesis system investigated in Chapter 9 also overcomes the effect of averaging across differing linguistic contexts, which causes decision tree regression to introduce degraded quality. The rich-context synthesis paradigm was found to produce highly natural and flexible speech and selects models which are trained in a constrained way, i.e., models are never calculated by averaging differing linguistic contexts. However, both rich-context synthesis and DNN synthesis systems are bound by the naturalness of vocoding. Throughout the investigations in Part I of the thesis, vocoding was found to dramatically reduce the quality of speech. However, it is assumed that as future improvements are made to vocoders, with model-able parameters, the performance of SPSS systems will track the improvements of the vocoder. Further investigation into the potential performance of rich-context HMM speech synthesis is of interest. This is of interest to gauge whether the use of neural network systems to guide the selection of models which are trained in a controlled way (i.e., using rich-context models) can outperform DNN-based systems which have less strict methods of training but are less transparent. This lack of transparency makes detection of issues with bad convergence of differing linguistic contexts within the neural network difficult to detect and fix.

On the other hand, using recorded units of speech, such as the synthesis systems investigated in Chapter 10, provides the ultimate synthesis naturalness. The investigation in Chapter 10 indicated that, within the hybrid synthesis paradigm, we are able to make use of the benefits of both the highly natural unit selection synthesis system and the SPSS system used to drive the selection of units. The investigation in Chapter 10 also found that subsequent improvements made to the underlying parametric system used to drive the hybrid system were reflected in hybrid synthesis performance. As such, future research in SPSS systems is of paramount importance, as these gains are likely to boost hybrid synthesis system performance as well as the SPSS system itself.

Bibliography

Alku, P. (1992). Glottal wave analysis with pitch synchronous iterative adaptive inverse filtering. *Speech Communication*, 11(2–3):109–118.

Banos, E., Erro, D., Bonafonte, A., and Moreno, A. (2008). Flexible harmonic/stochastic modeling for HMM-based speech synthesis. In *Proc. V Jornadas en Tecnologias del Habla*, pages 145–148.

Bengio, S. and Heigold, G. (2014). Word Embeddings for Speech Recognition. In *Proc. Interspeech*, number September, pages 1053–1057.

Black, A. W. (2006). CLUSTERGEN: a statistical parametric synthesizer using trajectory modeling. In *Proc. Interspeech*.

Black, A. W. and Muthukumar, P. K. (2015). Random forests for statistical speech synthesis. In *Proc. Interspeech*.

Black, A. W., Taylor, P., and Caley, R. (2001). The Festival speech synthesis system: system documentation. Technical report, University of Edinburgh.

Black, A. W. and Taylor, P. A. (1997). Automatically clustering similar units for unit selection in speech synthesis. In *Proc. Eurospeech*.

Borg, I. and Groenen, P. J. (2005). *Modern Multidimensional Scaling*. Springer.

Borg, I., Groenen, P. J., and Mair, P. (2013). *Applied multidimensional scaling*. Springer.

Cabral, J. P. S. R. (2011). *HMM-based Speech Synthesis Using an Acoustic Glottal Source Model*. PhD thesis, The University of Edinburgh.

Chalamandaris, A., Tsiakoulis, P., Karabetsos, S., and Raptis, S. (2013). The ILSP/INNOETICS text-to-speech system for the Blizzard Challenge 2013. In *Proc. Blizzard Challenge workshop*.

Chalamandaris, A., Tsiakoulis, P., Karabetsos, S., and Raptis, S. (2014). The ILSP/INNOETICS text-to-speech system for the Blizzard Challenge 2014. In *Proc. Blizzard Challenge workshop*.

Chen, L.-H., Ling, Z.-H., Jiang, Y., Song, Y., Xia, X.-J., Zu, Y.-Q., Yan, R.-Q., and Dai, L.-R. (2013). The USTC system for Blizzard Challenge 2013. In *Proc. Blizzard Challenge workshop*.

Chen, L.-H., Raitio, T., Valentini-Botinhao, C., Yamagishi, J., and Ling, Z.-H. (2014). DNN-based stochastic postfilter for HMM-based speech synthesis. In *Proc. Interspeech*, pages 1954–1958.

Chen, L.-H., Yang, C.-Y., Ling, Z.-H., Jiang, Y., Dai, L.-R., Hu, Y., and Wang, R.-H. (2011). The USTC system for Blizzard Challenge 2011. In *Proc. Blizzard Challenge workshop*.

Chen, S.-H., Hwang, S.-H., and Wang, Y.-R. (1998). An RNN-based prosodic information synthesizer for Mandarin text-to-speech. *IEEE Trans. Speech Audio Process.*, 6(3):226–239.

Clark, R. A., Richmond, K., and King, S. (2004). Festival 2–build your own general purpose unit selection speech synthesiser. In *Proc. SSW*.

Clark, R. A., Richmond, K., and King, S. (2007). Multisyn: Open-domain unit selection for the festival speech synthesis system. *Speech Communication*, 49(4):317–330.

Conkie, A. (1999). A robust unit selection system for speech synthesis. *The Journal of the Acoustical Society of America*, 105(2).

Conkie, A. D. and Isard, S. (1997). Optimal coupling of diphones. In *Progress in speech synthesis*, pages 293–304. Springer.

Cooke, M., Mayo, C., and Valentini-Botinhao, C. (2013a). Hurricane natural speech corpus, [sound]. LISTA Consortium, doi:10.7488/ds/140, 2013.

Cooke, M., Mayo, C., and Valentini-Botinhao, C. (2013b). Intelligibility-enhancing speech modifications: the Hurricane Challenge. In *Proc. Interspeech*, pages 2–6.

Donovan, R. E. and Woodland, P. C. (1995). Automatic speech synthesiser parameter estimation using HMMs. In *Proc. ICASSP*, pages 640–643. IEEE.

Drugman, T. and Dutoit, T. (2012). The deterministic plus stochastic model of the residual signal and its applications. *IEEE Audio, Speech, Language Process.*, 20(3).

Duda, R. O., Hart, P. E., and Stork, D. G. (2001). *Pattern classification*. John Wiley & Sons.

Erro, D., Moreno, A., and Bonafonte, A. (2007). Flexible harmonic/stochasic speech synthesis. In *Proc. SSW*, pages 194–199.

Erro, D., Sainz, I., Saratxaga, I., Navas, E., and Hernáez, I. (2010). MFCC+F0 extraction and waveform reconstruction using HNM: preliminary results in an HMM-based synthesizer. In *Proc. FALA*, pages 29–32.

Esquerra, I., Bonafonte, A., Vallverdú, F., and Febrer, A. (1998). A bilingual Spanish-Catalan database of units for concatenative synthesis. In *Workshop On Language Resources for European Minority Languages*.

Fan, Y., Qian, Y., Xie, F., and Soong, F. K. (2014). TTS synthesis with bidirectional LSTM based recurrent neural networks. In *Proc. Interspeech*, pages 1964–1968.

Fant, G. (1960). *Acoustic Theory of Speech Production*. Mouton, The Hague.

Fernandez, R., Rendel, A., Ramabhadran, B., and Hoory, R. (2015). Using Deep Bidirectional Recurrent Neural Networks for Prosodic-Target Prediction in a Unit-Selection Text-to-Speech System. In *Proc. Interspeech*.

Fukada, T., Tokuda, K., Kobayashi, T., and Imai, S. (1992). An adaptive algorithm for mel-cepstral analysis of speech. In *Proc. ICASSP*, pages 137–140.

Gales, M. and Young, S. (2008). The application of hidden Markov models in speech recognition. *Foundations and trends in signal processing*, 1(3):195–304.

Gales, M. J. F. (1999). Semi-tied covariance matrices for hidden Markov models. *IEEE Trans. Acoust., Speech, Signal Process.*, 7(3):272–281.

Hashimoto, K., Oura, K., Nankaku, Y., and Tokuda, K. (2015). The effect of neural networks in statistical parametric speech synthesis. In *Proc. ICASSP*, pages 4455–4459.

Henter, G. E., Merritt, T., Shannon, M., Mayo, C., and King, S. (2014a). Measuring the perceptual effects of modelling assumptions in speech synthesis using stimuli

constructed from repeated natural speech. In *Proc. Interspeech*, number September, pages 1504–1508.

Henter, G. E., Merritt, T., Shannon, M., Mayo, C., and King, S. (2014b). Repeated Harvard Speech corpus version 0.5, [dataset]. University of Edinburgh, The Centre for Speech Technology Research (CSTR); Cambridge University Engineering Department. doi:10.7488/ds/39.

Hershey, J. R. and Olsen, P. a. (2007). Approximating the Kullback-Leibler divergence between Gaussian mixture models. In *Proc. ICASSP*.

Hirai, T., Yamagishi, J., and Tenpaku, S. (2007). Utilization of an hmm-based feature generation module in 5 ms segment concatenative speech synthesis. In *Proc. SSW*, pages 81–84.

Hojo, N., Yoshizato, K., and Kameoka, H. (2013). Text-to-speech synthesizer based on combination of composite wavelet and hidden Markov models. In *Proc. SSW*, volume 2, pages 129–134.

Hu, Q., Stylianou, Y., Maia, R., Richmond, K., Yamagishi, J., and Latorre, J. (2014a). An investigation of the application of dynamic sinusoidal models to statistical parametric speech synthesis. In *Proc. Interspeech*, pages 780–784.

Hu, Q., Stylianou, Y., Richmond, K., Maia, R., Yamagishi, J., and Latorre, J. (2014b). A fixed dimension and perceptually based dynamic sinusoidal model of speech. In *Proc. ICASSP*, pages 6270–6274.

Hu, Y.-J. and Ling, Z.-H. (2016). DBN-based spectral feature representation for statistical parametric speech synthesis. *IEEE Signal Process. Lett.*, 23(3):321–325.

Hunt, A. J. and Black, A. W. (1996). Unit selection in a concatenative speech synthesis system using a large speech database. In *Proc. ICASSP*, pages 373–376.

IEEE (1969). IEEE recommended practice for speech quality measurement. *tae*, 17(3):225 – 246.

Imai, S. (1983). Cepstral analysis synthesis on the mel frequency scale. In *Proc. ICASSP*, pages 93–96.

Imai, S., Kobayashi, T., Tokuda, K., Masuko, T., Koishida, K., Sako, S., and Zen, H. (2012). Speech signal processing toolkit (SPTK), version 3.6.

ITU Recommendation ITU-R BS.1534-1 (2003). *Method for the subjective assessment of intermediate quality level of coding systems*. International Telecommunication Union Radiocommunication Assembly, Geneva, Switzerland.

ITU Recommendation ITU-T P.56 (2011). *Objective measurement of active speech level*. International Telecommunication Union, Telecommunication Standardization Sector, Geneva, Switzerland.

Jackson, P. J. B. and Shadle, C. (2001). Pitch-scaled estimation of simultaneous voiced and turbulence-noise components in speech. *IEEE Trans. Speech Audio Process.*, 9(7):713–726.

Kaszczuk, M. and Osowski, L. (2006). Evaluating Ivona speech synthesis system for Blizzard Challenge 2006. In *Proc. Blizzard Challenge workshop*.

Kaszczuk, M. and Osowski, L. (2007). The IVO software Blizzard 2007 entry: Improving Ivona speech synthesis system. In *Proc. Blizzard Challenge workshop*.

Kaszczuk, M. and Osowski, L. (2009). The IVO software Blizzard Challenge 2009 entry: Improving IVONA text-to-speech. In *Proc. Blizzard Challenge workshop*.

Kawahara, H. (2006). STRAIGHT, exploitation of the other aspect of VOCODER: Perceptually isomorphic decomposition of speech sounds. *Acoust. Sci. Technol.*, 27(6):349–353.

Kawahara, H., Estill, J., and Fujimura, O. (2001). Aperiodicity extraction and control using mixed mode excitation and group delay manipulation for a high quality speech analysis, modification and synthesis system STRAIGHT. In *2nd International Workshop on Models and Analysis of Vocal Emissions for Biomedical Applications (MAVEBA)*.

Kawahara, H., Masuda-Katsuse, I., and Cheveign, A. d. (1999). Restructuring speech representations using a pitch-adaptive time-frequency smoothing and an instantaneous-frequency-based F0 extraction : Possible role of a repetitive structure in sounds. *Speech Communication*, 27(3):187–207.

Kay, R., Watts, O., Chicote, R. B., and Mayo, C. (2015). Knowledge versus data in TTS: Evaluation of a continuum of synthesis systems. In *Proc. Interspeech*.

King, S. (2011). An introduction to statistical parametric speech synthesis. *Sadhana*, 36(October):837–852.

King, S. (2014). Measuring a decade of progress in text-to-speech. *Loquens*, 1(1).

King, S. and Karaiskos, V. (2009). The Blizzard Challenge 2009. In *Proc. Blizzard Challenge workshop*.

King, S. and Karaiskos, V. (2010). The Blizzard Challenge 2010. In *Proc. Blizzard Challenge workshop*.

King, S. and Karaiskos, V. (2011). The Blizzard Challenge 2011. In *Proc. Blizzard Challenge workshop*.

King, S. and Karaiskos, V. (2012). The Blizzard Challenge 2012. In *Proc. Blizzard Challenge workshop*.

King, S. and Karaiskos, V. (2013). The Blizzard Challenge 2013. In *Proc. Blizzard Challenge workshop*.

Koishida, K. (1998). *Low bit rate speech coding based on Mel-generalised cepstral analysis*. PhD thesis, Tokyo Institute of Technology.

Koishida, K., Tokuda, K., Kobayashi, T., and Imai, S. (1995). CELP coding based on Mel-cepstral analysis. In *Proc. ICASSP*, pages 33–36.

Kruskal, J. B. and Wish, M. (1978). *Multidimensional scaling*, volume 11. Sage.

Latorre, J., Gales, M. J., Buchholz, S., Knill, K., Tamura, M., Ohtani, Y., and Akamine, M. (2011). Continuous f0 in the source-excitation generation for HMM-based TTS : Do we need voiced / unvoiced classification? *Proc. ICASSP*, pages 4724–4727.

Liang, H., Qian, Y., Soong, F. K., and Liu, G. (2008). A cross-language state mapping approach to bilingual (Mandarin-English) TTS. In *Proc. ICASSP*, pages 4641–4644.

Ling, Z.-H., Kang, S.-Y., Zen, H., Senior, A., Schuster, M., Qian, X.-J., Meng, H. M., and Deng, L. (2015). Deep learning for acoustic modeling in parametric speech generation: A systematic review of existing techniques and future trends. *IEEE Signal Process. Mag.*, 32(3):35–52.

Ling, Z.-H., Lu, H., Hu, G.-P., Dai, L.-R., and Wang, R.-H. (2008). The USTC system for Blizzard Challenge 2008. In *Proc. Blizzard Challenge workshop*.

Ling, Z.-H. and Wang, R.-H. (2006). HMM-based unit selection using frame sized speech segments. In *Proc. Interspeech*, pages 2034–2037.

Ling, Z.-H. and Wang, R.-H. (2007). HMM-based hierarchical unit selection combining Kullback-Leibler divergence with likelihood criterion. In *Proc. ICASSP*, pages 1245–1248.

Ling, Z.-H. and Wang, R.-H. (2008). Minimum unit selection error training for HMM-based unit selection speech synthesis system. In *Proc. ICASSP*, pages 3949–3952.

Ling, Z.-H., Xia, X.-J., Song, Y., Yang, C.-Y., Chen, L.-H., and Dai, L.-R. (2012). The USTC system for Blizzard Challenge 2012. In *Proc. Blizzard Challenge workshop*.

Liu, C. and Kewley-Port, D. (2004). STRAIGHT: A new speech synthesizer for vowel formant discrimination. *Acoustics Research Letters Online*, 5(2):31.

MacKenzie, I. S. (2013). *Human-computer interaction: An empirical research perspective*. Elsevier.

Mayo, C., Clark, R. A. J., and King, S. (2005). Multidimensional scaling of listener responses to synthetic speech. In *Proc. Interspeech*, pages 2–5.

Mayo, C., Clark, R. A. J., and King, S. (2011). Listeners' weighting of acoustic cues to synthetic speech naturalness: A multidimensional scaling analysis. *Speech Communication*, 53(3):311–326.

Merritt, T., Clark, R. A. J., Wu, Z., Yamagishi, J., and King, S. (2016a). Deep neural network-guided unit selection synthesis. In *Proc. ICASSP*.

Merritt, T., Clark, R. A. J., Wu, Z., Yamagishi, J., and King, S. (2016b). Listening test materials for "Deep neural network-guided unit selection synthesis", 2016 [dataset]. University of Edinburgh, The Centre for Speech Technology Research (CSTR), doi:10.7488/ds/1313.

Merritt, T. and King, S. (2013). Investigating the shortcomings of HMM synthesis. In *Proc. SSW*, pages 165–170.

Merritt, T., Latorre, J., and King, S. (2015a). Attributing modelling errors in HMM synthesis by stepping gradually from natural to modelled speech. In *Proc. ICASSP*.

Merritt, T., Raitio, T., and King, S. (2014). Investigating source and filter contributions, and their interaction, to statistical parametric speech synthesis. In *Proc. Interspeech*, number September, pages 1509–1513.

Merritt, T., Ronanki, S., Wu, Z., and Watts, O. (2016c). The CSTR entry to the Blizzard Challenge 2016. In *Proc. Blizzard Challenge workshop*.

Merritt, T., Yamagishi, J., Wu, Z., Watts, O., and King, S. (2015b). Deep neural network context embeddings for model selection in rich-context HMM synthesis. In *Proc. Interspeech*.

Merritt, T., Yamagishi, J., Wu, Z., Watts, O., and King, S. (2015c). Listening test materials for "Deep neural network context embeddings for model selection in rich-context HMM synthesis", 2015 [dataset]. University of Edinburgh, The Centre for Speech Technology Research (CSTR), doi:10.7488/ds/256.

Morgan, N., Cohen, J., Parthasarathi, S. H. K., Chang, S.-Y., and Wegmann, S. (2013). Final Report: OUCH Project (Outing Unfortunate Characteristics of HMMs). Technical report.

Moulines, E. and Charpentier, F. (1990). Pitch-synchronous waveform processing techniques for text-to-speech synthesis using diphones. *Speech Communication*, 9:453–467.

Odell, J. J. (1995). *The use of context in large vocabulary speech recognition*. PhD thesis, Cambridge University.

Pollet, V. and Breen, A. (2008). Synthesis by generation and concatenation of multiform segments. In *Proc. Interspeech*, pages 1825–1828.

Qian, Y., Soong, F. K., and Yan, Z.-J. (2013). A unified trajectory tiling approach to high quality speech rendering. *IEEE/ACM Trans. Audio, Speech, Language Process.*, 21(2):280–290.

Qian, Y., Yan, Z.-j., Wu, Y., Soong, F., Zhuang, X., and Kong, S. (2010). An HMM Trajectory Tiling (HTT) Approach to High Quality TTS. In *Proc. Interspeech*, number September, pages 422–425.

Raitio, T., Suni, A., Pulakka, H., Vainio, M., and Alku, P. (2010). Comparison of formant enhancement methods for HMM-based speech synthesis. In *Proc. SSW*, pages 1–6.

Raitio, T., Suni, A., Pulakka, H., Vainio, M., and Alku, P. (2011a). Utilizing glottal source pulse library for generating improved excitation signal for HMM-based speech synthesis. In *Proc. ICASSP*, pages 4564–4567.

Raitio, T., Suni, A., Vainio, M., and Alku, P. (2013). Comparing glottal-flow-excited statistical parametric speech synthesis methods. In *Proc. ICASSP*, pages 7830–7834.

Raitio, T., Suni, A., Vainio, M., and Alku, P. (2014). Synthesis and perception of breathy, normal, and lombard speech in the presence of noise. *Computer Speech & Language*, 28(2):648–664.

Raitio, T., Suni, A., Yamagishi, J., Pulakka, H., Nurminen, J., Vainio, M., and Alku, P. (2011b). HMM-based speech synthesis utilizing glottal inverse filtering. *IEEE/ACM Trans. Audio, Speech, Language Process.*, 19(1):153–165.

Ribeiro, M. S. and Clark, R. A. J. (2015). A multi-level representation of f0 using the continuous wavelet transform and the discrete cosine transform. In *Proc. ICASSP*, pages 4909–4913.

Ribeiro, M. S., Watts, O., Yamagishi, J., and Clark, R. A. J. (2016). A multi-level representation of f0 using the continuous wavelet transform and the discrete cosine transform. In *Proc. ICASSP*, pages 5525–5529.

Ribeiro, M. S., Yamagishi, J., and Clark, R. A. J. (2015). A perceptual investigation of wavelet-based decomposition of f0 for text-to-speech synthesis. In *Proc. Interspeech*.

Richmond, K., Hoole, P., and King, S. (2011). Announcing the Electromagnetic Articulography (Day 1) Subset of the mngu0 Articulatory Corpus. In *Proc. Interspeech*, number August, pages 1505–1508.

Richmond, K. and King, S. (2016). Smooth talking: articulatory join costs for unit selection. In *Proc. ICASSP*.

Shannon, M., Zen, H., and Byrne, W. (2011). The Effect of Using Normalized Models in Statistical Speech Synthesis. In *Proc. Interspeech*, number 1, pages 1–4.

Shinoda, K. and Watanabe, T. (2000). MDL-based context-dependent subword modeling for speech recognition. *The Journal of the Acoustical Society of Japan (E)*, 21(2):79–86.

Silén, H. and Helander, E. (2012). Ways to Implement Global Variance in Statistical Speech Synthesis. In *Proc. Interspeech*, pages 1436–1439.

Sorin, A., Shechtman, S., and Pollet, V. (2011). Uniform Speech Parameterization for Multi-form Segment Synthesis. In *Proc. Interspeech*, number August, pages 337–340.

Sorin, A., Shechtman, S., and Pollet, V. (2012). Psychoacoustic Segment Scoring for Multi-Form Speech Synthesis. In *Proc. Interspeech*, number 3, pages 2214–2217.

Sorin, A., Shechtman, S., and Pollet, V. (2014). Refined Inter-segment Joining in Multi-Form Speech Synthesis. In *Proc. Interspeech*, number September, pages 790–794.

Stylianou, I. (1996). *Harmonic plus noise models for speech, combined with statistical methods, for speech and speaker modification*. PhD thesis, École Nationale Supérieure des Télécommunications.

Stylianou, Y. (2001). Applying the harmonic plus noise model in concatenative speech synthesis. *IEEE Trans. Speech Audio Process.*, 9(1):21–29.

Suendermann, D., Höge, H., and Black, A. (2010). Challenges in speech synthesis. *Speech Technology*, pages 19–32.

Sündermann, D., Strecha, G., Bonafonte, A., Höge, H., and Ney, H. (2005). Evaluation of VTLN-based voice conversion for embedded speech synthesis. In *Proc. Interspeech*, pages 2593–2596.

Suni, A., Aalto, D., Raitio, T., Alku, P., and Vainio, M. (2013). Wavelets for intonation modeling in hmm speech synthesis. In *Proc. SSW*.

Suni, A., Raitio, T., Vainio, M., and Alku, P. (2010). The GlottHMM speech synthesis entry for Blizzard Challenge 2010. In *Proc. Blizzard Challenge workshop*. http://festvox.org/blizzard.

Suni, A., Raitio, T., Vainio, M., and Alku, P. (2011). The GlottHMM entry for Blizzard Challenge 2011: Utilizing source unit selection in HMM-based speech synthesis for improved excitation generation. In *Proc. Blizzard Challenge workshop*.

Suni, A., Raitio, T., Vainio, M., and Alku, P. (2012). The GlottHMM Entry for Blizzard Challenge 2012: Hybrid Approach. In *Proc. Blizzard Challenge workshop*.

Takaki, S., Kim, S., Yamagishi, J., and Kim, J. (2015). Multiple feed-forward deep neural networks for statistical parametric speech synthesis. In *Proc. Interspeech*.

Takamichi, S., Toda, T., Black, A. W., and Nakamura, S. (2014a). Modified post-filter to recover modulation spectrum for HMM-based speech synthesis. In *GlobalSIP*, pages 710–714.

Takamichi, S., Toda, T., Black, A. W., and Nakamura, S. (2015). Parameter generation algorithm considering modulation spectrum for HMM-based speech synthesis. In *Proc. ICASSP*, pages 4210–4214.

Takamichi, S., Toda, T., Neubig, G., Sakti, S., and Nakamura, S. (2014b). A postfilter to modify the modulation spectrum in HMM-based speech synthesis. In *Proc. ICASSP*, pages 290–294.

Takamichi, S., Toda, T., Shiga, Y., Kawai, H., Sakti, S., and Nakamura, S. (2012). An evaluation of parameter generation methods with rich context models in HMM-based speech synthesis. In *Proc. Interspeech*, pages 1139–1142.

Takamichi, S., Toda, T., Shiga, Y., and Sakti, S. (2013). Improvements to HMM-Based Speech Synthesis Based on Parameter Generation with Rich Context Models. In *Proc. Interspeech*, number August, pages 364–368.

Takamichi, S., Toda, T., Shiga, Y., Sakti, S., Neubig, G., Nakamura, S., and Member, S. (2014c). Parameter generation methods with rich context models for high-quality and flexible text-to-speech synthesis. *IEEE J. Sel. Topics Signal Process.*, 8(2):239–250.

Tao, J., Hirose, K., Tokuda, K., Black, A., and King, S. (2014). Introduction to the issue on statistical parametric speech synthesis. *IEEE J. Sel. Topics Signal Process.*, 8(2):170–172.

Taylor, P. (2006a). The target cost formulation in unit selection speech synthesis. In *Proc. Interspeech*, pages 2038–2041.

Taylor, P. (2006b). Unifying unit selection and hidden Markov model speech synthesis. In *Proc. Interspeech*, pages 1758–1761.

Taylor, P. (2009). *Text-to-speech synthesis*. Cambridge university press.

Taylor, P., Black, A. W., and Caley, R. (1998). The architecture of the festival speech synthesis system. In *The Third ESCA/COCOSDA Workshop on Speech Synthesis*.

Titze, I. R. (2008). Nonlinear source-filter coupling in phonation: Theory. *JASA*, 123(5):2733–2749.

Toda, T. and Tokuda, K. (2007). A Speech Parameter Generation Algorithm Considering Global Variance for HMM-Based Speech Synthesis. *IEICE Transactions on Information and Systems*, E90-D(5):816–824.

Tokuda, K., Kobayashi, T., Masuko, T., and Imai, S. (1994). Mel-generalized cepstral analysis – a unified approach to speech spectral estimation. In *ICSLP*.

Tokuda, K., Masuko, T., Miyazaki, N., and Kobayashi, T. (2002). Multi-space probability distribution HMM. *IEICE Transactions on Information and Systems*, 85(3):455–464.

Tokuda, K., Nankaku, Y., Toda, T., Zen, H., Yamagishi, J., and Oura, K. (2013). Speech Synthesis Based on Hidden Markov Models. *Proceedings of the IEEE*, 101(5):1234–1252.

Tokuda, K., Yoshimura, T., Masuko, T., Kobayashi, T., and Kitamura, T. (2000). Speech parameter generation algorithms for HMM-based speech synthesis. *Proc. ICASSP*, (3):2–5.

Tokuda, K. and Zen, H. (2015). Directly modeling speech waveforms by neural networks for statistical parametric speech synthesis. In *Proc. ICASSP*, pages 4215–4219.

Tokuda, K. and Zen, H. (2016). Directly modeling voiced and unvoiced components in speech waveforms by neural networks. In *Proc. ICASSP*, pages 5640–5644.

Valentini-Botinhao, C. (2013). *Intelligibility enhancement of synthetic speech in noise*. PhD thesis, University of Edinburgh.

Veaux, C., Yamagishi, J., and King, S. (2012). Using HMM-based speech synthesis to reconstruct the voice of individuals with degenerative speech disorders. In *Proc. Interspeech*, pages 967–970.

Vepa, J. and King, S. (2006). Subjective evaluation of join cost and smoothing methods for unit selection speech synthesis. *IEEE Audio, Speech, Language Process.*, 14(5):1763–1771.

Wan, V., Latorre, J., Yanagisawa, K., Braunschweilers, N., Chen, L., Gales, M., and Akamine, M. (2014). Building HMM-TTS models on diverse data. *IEEE Journal of Selected Topics in Signal Processing*, 8(2).

Watts, O. (2012). *Unsupervised learning for text-to-speech synthesis*. PhD thesis, The University of Edinburgh.

Watts, O., Henter, G. E., Merritt, T., Wu, Z., and King, S. (2016a). From HMMs to DNNs: Where do the improvements come from? In *Proc. ICASSP*.

Watts, O., Henter, G. E., Merritt, T., Wu, Z., and King, S. (2016b). Listening test materials for "From HMMs to DNNs: Where do the improvements come from?", 2016 [dataset]. University of Edinburgh, The Centre for Speech Technology Research (CSTR), doi:10.7488/ds/1316.

Wester, M., Valentini-Botinhao, C., and Henter, G. E. (2015). Are we using enough listeners? No! an empirically-supported critique of Interspeech 2014 TTS evaluations. In *Proc. Interspeech*.

Wu, Z. and King, S. (2015). Minimum trajectory error training for deep neural networks, combined with stacked bottleneck features. In *Proc. Interspeech*.

Wu, Z., Valentini-Botinhao, C., Watts, O., and King, S. (2015). Deep neural networks employing multi-task learning and stacked bottleneck features for speech synthesis. In *Proc. ICASSP*.

Xia, X.-J., Ling, Z.-H., Jiang, Y., and Dai, L.-R. (2014). HMM-based unit selection speech synthesis using log likelihood ratios derived from perceptual data. *Speech Communication*, 63:27–37.

Yamagishi, J. (2006). An introduction to HMM-based speech synthesis. Technical report, Tokyo Institute of Technology.

Yamagishi, J., Ling, Z.-H., and King, S. (2008). Robustness of HMM-based speech synthesis. In *Proc. Interspeech*.

Yan, Z.-J., Qian, Y., and Soong, F. K. (2009). Rich context modeling for high quality HMM-based TTS. In *Proc. Interspeech*, pages 1755–1758.

Yan, Z.-J., Qian, Y., and Soong, F. K. (2010). Rich-context unit selection (RUS) approach to high quality TTS. In *Proc. ICASSP*, pages 4798–4801.

Yoshimura, T. (2002). *Simultaneous modeling of phonetic and prosodic parameters, and characteristic conversion for hmm-based text-to-speech systems*. PhD thesis, Nagoya Institute of Technology.

Yoshimura, T., Tokuda, K., Masuko, T., Kobayashi, T., and Kitamura, T. (1998). Duration modeling for HMM-based speech synthesis. In *ICSLP*, volume 98, pages 29–31.

Yoshimura, T., Tokuda, K., Masuko, T., Kobayashi, T., and Kitamura, T. (1999). Simultaneous modeling of spectrum, pitch and duration in HMM-based speech synthesis. In *Proc. Eurospeech*, pages 2347–2350.

Zen, H. (2015). Acoustic Modeling in Statistical Parametric Speech Synthesis - From HMM to LSTM-RNN. In *Proc. MLSLP*.

Zen, H. and Gales, M. (2011). Decision tree-based context clustering based on cross validation and hierarchical priors. In *Proc. ICASSP*, pages 4560–4563.

Zen, H., Nose, T., Yamagishi, J., Sako, S., Masuko, T., Black, A. W., and Tokuda, K. (2007a). The HMM-based speech synthesis system (HTS) version 2.0. *Proc. SSW*, pages 294–299.

Zen, H. and Sak, H. (2015). Unidirectional long short-term memory recurrent neural network with recurrent output layer for low-latency speech synthesis. In *Proc. ICASSP*, pages 4470–4474.

Zen, H. and Senior, A. (2014). Deep mixture density networks for acoustic modeling in statistical parametric speech synthesis. In *Proc. ICASSP*, pages 3872–3876.

Zen, H., Senior, A., and Schuster, M. (2013). Statistical parametric speech synthesis using deep neural networks. In *Proc. ICASSP*, pages 7962–7966.

Zen, H. and Toda, T. (2005). An overview of Nitech HMM-based speech synthesis system for Blizzard challenge 2005. In *Proc. Interspeech*, pages 93–96.

Zen, H., Tokuda, K., and Black, A. W. (2009). Statistical parametric speech synthesis. *Speech Communication*, 51(11):1039–1064.

Zen, H., Tokuda, K., Masuko, T., Kobayashi, T., and Kitamura, T. (2004). Hidden semi-Markov model based speech synthesis. In *Proc. Interspeech*.

Zen, H., Tokuda, K., Masuko, T., Kobayasih, T., and Kitamura, T. (2007b). A hidden semi-Markov model-based speech synthesis system. *IEICE transactions on information and systems*, 90(5):825–834.